LinuxQMISDK-Lite

SLQS04.00.07

Generated by Doxygen 1.8.6

Wed May 24 2017 06:42:04

Contents

1 Welcome to the Sierra Wireless Linux QMI SDK API Reference Guide 1

1.1 Important Notice . . . 1
1.2 Limitation of Liability . . . 1
1.3 Patents . . . 1
1.4 Copyright . . . 2
1.5 Trademarks . . . 2
1.6 Contact Information . . . 2

2 Module Index 3

2.1 Modules . . . 3

3 Namespace Index 5

3.1 Namespace List . . . 5

4 Data Structure Index 7

4.1 Data Structures . . . 7

5 File Index 23

5.1 File List . . . 23

6 Module Documentation 25

6.1 QMI pack/unpack (lite-qmi) . . . 25
6.1.1 Detailed Description . . . 25
6.2 Streaming Download Protocol (lite-fw) . . . 26
6.2.1 Detailed Description . . . 26

7 Namespace Documentation 27

7.1 Tables Namespace Reference . . . 27
7.1.1 Detailed Description . . . 27

8 Data Structure Documentation 29

8.1 _litefw_FirmwareInfo_ Struct Reference . . . 29
8.1.1 Detailed Description . . . 29
8.1.2 Field Documentation . . . 29

8.1.2.1 szCarrier_str . . . 30
8.1.2.2 szCarrierPriversion_str . . . 30
8.1.2.3 szFwversion_str . . . 30
8.1.2.4 szModelid_str . . . 30
8.1.2.5 szPackageid_str . . . 30
8.1.2.6 szSku_str . . . 30
8.2 _PackSwiAvmsSetSettingsAPNInfo Struct Reference . . . 30
8.2.1 Detailed Description . . . 30
8.2.2 Field Documentation . . . 30
8.2.2.1 bAPNLength . . . 30
8.2.2.2 bPWDLength . . . 30
8.2.2.3 bUnameLength . . . 30
8.2.2.4 szAPN . . . 30
8.2.2.5 szPWD . . . 30
8.2.2.6 szUname . . . 30
8.3 _PackSwiAvmsSetSettingsConnectionRetryTimers Struct Reference . . . 30
8.3.1 Field Documentation . . . 31
8.3.1.1 Timers . . . 31
8.4 _PackSwiAvmsSetSettingsPeriodInfo Struct Reference . . . 31
8.4.1 Detailed Description . . . 31
8.4.2 Field Documentation . . . 31
8.4.2.1 ulMax . . . 31
8.4.2.2 ulMin . . . 31
8.5 _PackSwiAVMSSettingsAPNInfo Struct Reference . . . 31
8.5.1 Field Documentation . . . 31
8.5.1.1 bAPNLength . . . 31
8.5.1.2 bPWDLength . . . 31
8.5.1.3 bUnameLength . . . 31
8.5.1.4 pAPN . . . 31
8.5.1.5 pPWD . . . 31
8.5.1.6 pUname . . . 31
8.6 _PackSwiAVMSSettingsConnectionRetryTimers Struct Reference . . . 31
8.6.1 Field Documentation . . . 32
8.6.1.1 Timers . . . 32
8.7 _PackSwiAVMSSettingsPeriodsInfo Struct Reference . . . 32
8.7.1 Field Documentation . . . 32
8.7.1.1 max . . . 32
8.7.1.2 min . . . 32
8.8 _unpack_swiavms_SLQSAVMSEventReportInd_t Struct Reference . . . 32
8.8.1 Detailed Description . . . 32

8.8.2 Field Documentation . . . 33
8.8.2.1 pBinaryUpdateSessionInfo . . . 33
8.8.2.2 pConfig . . . 33
8.8.2.3 pConnectionRequest . . . 33
8.8.2.4 pDataSessionStatus . . . 33
8.8.2.5 pHTTPStatus . . . 33
8.8.2.6 pNotification . . . 33
8.8.2.7 pPackageID . . . 33
8.8.2.8 pRegStatus . . . 33
8.8.2.9 pSessionType . . . 33
8.8.2.10 pWAMSParaChanged . . . 33
8.8.2.11 resultcode . . . 33
8.9 _unpack_swiavms_SLQSAVMSSessionGetInfo_t Struct Reference . . . 33
8.9.1 Detailed Description . . . 34
8.9.2 Field Documentation . . . 34
8.9.2.1 pBinaryUpdateSessionInfo . . . 34
8.9.2.2 pConfig . . . 34
8.9.2.3 pNotification . . . 34
8.9.2.4 pPackageID . . . 34
8.9.2.5 resultcode . . . 34
8.10 _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo Struct Reference . . . 34
8.10.1 Detailed Description . . . 34
8.10.2 Field Documentation . . . 35
8.10.2.1 bBinaryType . . . 35
8.10.2.2 bSerity . . . 35
8.10.2.3 bState . . . 35
8.10.2.4 bUserInputRequest . . . 35
8.10.2.5 szDescription . . . 35
8.10.2.6 szName . . . 36
8.10.2.7 szVersion . . . 36
8.10.2.8 TlvPresent . . . 36
8.10.2.9 ulPkgDownloadComplete . . . 36
8.10.2.10 ulPkgDownloadSize . . . 36
8.10.2.11 wDescriptionLength . . . 36
8.10.2.12 wNameLength . . . 36
8.10.2.13 wUpdateCompeteStatus . . . 36
8.10.2.14 wUserInputTimeout . . . 36
8.10.2.15 wVersionLength . . . 36
8.11 _UnpackSwiAvmsEventReportConfig Struct Reference . . . 36
8.11.1 Detailed Description . . . 36

8.11.2 Field Documentation . . . 36
8.11.2.1 bState . . . 37
8.11.2.2 bUserInputRequest . . . 37
8.11.2.3 szAlertMsg . . . 37
8.11.2.4 TlvPresent . . . 37
8.11.2.5 wAlertMsgLength . . . 37
8.11.2.6 wUserInputTimeout . . . 37
8.12 _UnpackSwiAvmsEventReportConnectionRequest Struct Reference . . . 37
8.12.1 Detailed Description . . . 37
8.12.2 Field Documentation . . . 37
8.12.2.1 bUserInputRequest . . . 37
8.12.2.2 TlvPresent . . . 37
8.12.2.3 wUserInputTimeout . . . 37
8.13 _UnpackSwiAvmsEventReportDataSessionStatus Struct Reference . . . 37
8.13.1 Detailed Description . . . 37
8.13.2 Field Documentation . . . 38
8.13.2.1 bType . . . 38
8.13.2.2 TlvPresent . . . 38
8.13.2.3 wErrorCode . . . 38
8.14 _UnpackSwiAvmsEventReportHTTPStatus Struct Reference . . . 38
8.14.1 Detailed Description . . . 38
8.14.2 Field Documentation . . . 39
8.14.2.1 TlvPresent . . . 39
8.14.2.2 wHTTPStatus . . . 40
8.15 _UnpackSwiAvmsEventReportNotification Struct Reference . . . 40
8.15.1 Detailed Description . . . 40
8.15.2 Field Documentation . . . 40
8.15.2.1 bNotification . . . 40
8.15.2.2 TlvPresent . . . 41
8.15.2.3 wSessionStatus . . . 41
8.16 _UnpackSwiAvmsEventReportPackageID Struct Reference . . . 41
8.16.1 Detailed Description . . . 41
8.16.2 Field Documentation . . . 41
8.16.2.1 bPackageID . . . 41
8.16.2.2 TlvPresent . . . 41
8.17 _UnpackSwiAvmsEventReportRegStatus Struct Reference . . . 41
8.17.1 Detailed Description . . . 41
8.17.2 Field Documentation . . . 41
8.17.2.1 bRegStatus . . . 41
8.17.2.2 TlvPresent . . . 41

8.18 _UnpackSwiAvmsEventReportSessionType Struct Reference . . . 41
8.18.1 Detailed Description . . . 42
8.18.2 Field Documentation . . . 42
8.18.2.1 bType . . . 42
8.18.2.2 TlvPresent . . . 42
8.19 _UnpackSwiAvmsEventReportWAMSParamChange Struct Reference . . . 42
8.19.1 Detailed Description . . . 42
8.19.2 Field Documentation . . . 42
8.19.2.1 TlvPresent . . . 42
8.19.2.2 wWamsChangeMask . . . 42
8.20 altSrcInfo_t Struct Reference . . . 42
8.20.1 Detailed Description . . . 43
8.20.2 Field Documentation . . . 43
8.20.2.1 coverage . . . 43
8.20.2.2 linkage . . . 43
8.20.2.3 source . . . 43
8.21 appStats Struct Reference . . . 43
8.21.1 Detailed Description . . . 44
8.21.2 Field Documentation . . . 46
8.21.2.1 aidLength . . . 46
8.21.2.2 aidVal . . . 46
8.21.2.3 appState . . . 46
8.21.2.4 appType . . . 46
8.21.2.5 persoFeature . . . 46
8.21.2.6 persoRetries . . . 46
8.21.2.7 persoState . . . 46
8.21.2.8 persoUnblockRetries . . . 46
8.21.2.9 pin1Retries . . . 46
8.21.2.10 pin1State . . . 46
8.21.2.11 pin2Retries . . . 46
8.21.2.12 pin2State . . . 46
8.21.2.13 puk1Retries . . . 46
8.21.2.14 puk2Retries . . . 46
8.21.2.15 univPin . . . 46
8.22 CarrierImage_t Struct Reference . . . 46
8.22.1 Detailed Description . . . 47
8.22.2 Field Documentation . . . 47
8.22.2.1 m_FwBuildId . . . 47
8.22.2.2 m_FwImageId . . . 47
8.22.2.3 m_nCarrierId . . . 47

8.22.2.4 m_nFolderId . . . 47
8.22.2.5 m_nStorage . . . 47
8.22.2.6 m_PriBuildId . . . 47
8.22.2.7 m_PriImageId . . . 48
8.23 cat_AlPhaIdentifierTlv Struct Reference . . . 48
8.23.1 Detailed Description . . . 48
8.23.2 Field Documentation . . . 48
8.23.2.1 AlphaID . . . 48
8.23.2.2 AlphaIDLength . . . 48
8.23.2.3 ReferenceID . . . 48
8.24 cat_commonEventTlv Struct Reference . . . 48
8.24.1 Detailed Description . . . 48
8.24.2 Field Documentation . . . 48
8.24.2.1 CatEvent . . . 49
8.24.2.2 EventID . . . 49
8.24.2.3 EventLength . . . 49
8.24.2.4 TlvPresent . . . 49
8.25 cat_currentCatEvent Union Reference . . . 49
8.25.1 Detailed Description . . . 49
8.25.2 Field Documentation . . . 49
8.25.2.1 CatAlphaIdtfr . . . 49
8.25.2.2 CatEndPS . . . 49
8.25.2.3 CatEventLst . . . 49
8.25.2.4 CatEvIDData . . . 49
8.25.2.5 CatRefresh . . . 49
8.26 cat_EndProactiveSessionTlv Struct Reference . . . 49
8.26.1 Detailed Description . . . 50
8.26.2 Field Documentation . . . 50
8.26.2.1 EndProactiveSession . . . 50
8.27 cat_EventIDDataTlv Struct Reference . . . 50
8.27.1 Detailed Description . . . 50
8.27.2 Field Documentation . . . 50
8.27.2.1 Data . . . 50
8.27.2.2 DataLength . . . 50
8.27.2.3 ReferenceID . . . 50
8.28 cat_EventListTlv Struct Reference . . . 50
8.28.1 Detailed Description . . . 51
8.28.2 Field Documentation . . . 51
8.28.2.1 SetupEventList . . . 51
8.29 cat_RefreshTlv Struct Reference . . . 51

8.29.1 Detailed Description . . . 51
8.29.2 Field Documentation . . . 51
8.29.2.1 RefreshMode . . . 51
8.29.2.2 RefreshStage . . . 51
8.30 cdmaSSInfo Struct Reference . . . 51
8.30.1 Detailed Description . . . 52
8.30.2 Field Documentation . . . 52
8.30.2.1 ecio . . . 52
8.30.2.2 rssi . . . 52
8.31 connectionStatus Struct Reference . . . 52
8.31.1 Detailed Description . . . 52
8.31.2 Field Documentation . . . 52
8.31.2.1 MDMCallDuration . . . 52
8.31.2.2 MDMConnStatus . . . 52
8.32 crashInfoParams Struct Reference . . . 53
8.32.1 Detailed Description . . . 53
8.32.2 Field Documentation . . . 53
8.32.2.1 crashInfo . . . 53
8.32.2.2 crashStatus . . . 53
8.33 crashInformation Struct Reference . . . 53
8.33.1 Detailed Description . . . 53
8.33.2 Field Documentation . . . 54
8.33.2.1 crashData . . . 54
8.33.2.2 crashId . . . 54
8.33.2.3 crashString . . . 54
8.33.2.4 crashStrlen . . . 54
8.33.2.5 gcdumpString . . . 54
8.33.2.6 gcdumpStrlen . . . 54
8.33.2.7 numCrashes . . . 54
8.34 currNetworkInfo Struct Reference . . . 54
8.34.1 Detailed Description . . . 54
8.34.2 Field Documentation . . . 55
8.34.2.1 NetworkType . . . 56
8.34.2.2 RATMask . . . 56
8.34.2.3 SOMask . . . 56
8.35 dms_ActivationStatusTlv Struct Reference . . . 56
8.35.1 Detailed Description . . . 56
8.35.2 Field Documentation . . . 56
8.35.2.1 activationStatus . . . 56
8.35.2.2 TlvPresent . . . 56

8.36 dms_OperatingModeTlv Struct Reference . . . 56
8.36.1 Detailed Description . . . 57
8.36.2 Field Documentation . . . 57
8.36.2.1 operatingMode . . . 57
8.36.2.2 TlvPresent . . . 57
8.37 DMScustSettingInfo Struct Reference . . . 57
8.37.1 Detailed Description . . . 57
8.37.2 Field Documentation . . . 58
8.37.2.1 cust_attr . . . 58
8.37.2.2 cust_id . . . 58
8.37.2.3 cust_value . . . 58
8.37.2.4 id_length . . . 58
8.37.2.5 value_length . . . 58
8.38 DMScustSettingList Struct Reference . . . 58
8.38.1 Detailed Description . . . 58
8.38.2 Field Documentation . . . 58
8.38.2.1 custSetting . . . 58
8.38.2.2 list_type . . . 59
8.38.2.3 num_instances . . . 59
8.39 DMSgetCustomFeatureV2 Struct Reference . . . 59
8.39.1 Detailed Description . . . 59
8.39.2 Field Documentation . . . 59
8.39.2.1 pCustSettingInfo . . . 59
8.39.2.2 pCustSettingList . . . 59
8.39.2.3 pGetCustomInput . . . 59
8.40 DMSgetCustomInput Struct Reference . . . 59
8.40.1 Detailed Description . . . 59
8.40.2 Field Documentation . . . 60
8.40.2.1 cust_id . . . 60
8.40.2.2 list_type . . . 60
8.41 dunchannelRate Struct Reference . . . 60
8.41.1 Detailed Description . . . 60
8.41.2 Field Documentation . . . 60
8.41.2.1 CurrChanRxRate . . . 60
8.41.2.2 CurrChanTxRate . . . 60
8.41.2.3 MaxChanRxRate . . . 60
8.41.2.4 MaxChanTxRate . . . 60
8.42 eriDataparams Struct Reference . . . 60
8.42.1 Detailed Description . . . 61
8.42.2 Field Documentation . . . 61

8.42.2.1 eriData . . . 61
8.42.2.2 eriDataLen . . . 61
8.43 eTWSPLMNInfoTlv Struct Reference . . . 61
8.43.1 Detailed Description . . . 61
8.43.2 Field Documentation . . . 61
8.43.2.1 ETWSPLMNInfo . . . 61
8.43.2.2 TlvPresent . . . 61
8.44 FMSImageElement Struct Reference . . . 61
8.44.1 Detailed Description . . . 62
8.44.2 Field Documentation . . . 62
8.44.2.1 buildId . . . 62
8.44.2.2 buildIdLength . . . 62
8.44.2.3 imageId . . . 62
8.44.2.4 imageType . . . 62
8.45 FMSImageIdElement Struct Reference . . . 62
8.45.1 Detailed Description . . . 62
8.45.2 Field Documentation . . . 63
8.45.2.1 buildID . . . 63
8.45.2.2 buildIDLength . . . 63
8.45.2.3 failureCount . . . 63
8.45.2.4 imageID . . . 63
8.45.2.5 storageIndex . . . 63
8.46 FMSImageIDEntries Struct Reference . . . 63
8.46.1 Detailed Description . . . 63
8.46.2 Field Documentation . . . 64
8.46.2.1 executingImage . . . 64
8.46.2.2 imageIDElement . . . 64
8.46.2.3 imageIDSize . . . 64
8.46.2.4 imageType . . . 64
8.46.2.5 maxImages . . . 64
8.47 FMSImageList Struct Reference . . . 64
8.47.1 Detailed Description . . . 64
8.47.2 Field Documentation . . . 64
8.47.2.1 imageIDEntries . . . 64
8.47.2.2 listSize . . . 64
8.48 FMSPrefImageList Struct Reference . . . 65
8.48.1 Detailed Description . . . 65
8.48.2 Field Documentation . . . 65
8.48.2.1 listEntries . . . 65
8.48.2.2 listSize . . . 65

8.49 hdrSSInfo Struct Reference . . . 65
8.49.1 Detailed Description . . . 65
8.49.2 Field Documentation . . . 66
8.49.2.1 ecio . . . 66
8.49.2.2 io . . . 66
8.49.2.3 rssi . . . 66
8.49.2.4 sinr . . . 66
8.50 image_info_t Struct Reference . . . 66
8.50.1 Detailed Description . . . 66
8.50.2 Field Documentation . . . 67
8.50.2.1 buildID . . . 67
8.50.2.2 buildIDLen . . . 67
8.50.2.3 imageType . . . 67
8.50.2.4 uniqueID . . . 67
8.51 ipv6AddressInfo Struct Reference . . . 67
8.51.1 Detailed Description . . . 67
8.51.2 Field Documentation . . . 67
8.51.2.1 IPAddressV6 . . . 67
8.51.2.2 IPV6PrefixLen . . . 67
8.52 LibPackGPRSRequestedQoS Struct Reference . . . 67
8.52.1 Detailed Description . . . 68
8.52.2 Field Documentation . . . 68
8.52.2.1 delayClass . . . 68
8.52.2.2 meanThroughputClass . . . 68
8.52.2.3 peakThroughputClass . . . 68
8.52.2.4 precedenceClass . . . 68
8.52.2.5 reliabilityClass . . . 68
8.53 LibpackProfile3GPP Struct Reference . . . 68
8.53.1 Detailed Description . . . 69
8.53.2 Field Documentation . . . 73
8.53.2.1 pAddrAllocPref . . . 73
8.53.2.2 pAPNClass . . . 73
8.53.2.3 pAPNDisabledFlag . . . 73
8.53.2.4 pAPNName . . . 73
8.53.2.5 pAPNnameSize . . . 73
8.53.2.6 pAuthenticationPref . . . 73
8.53.2.7 pGPRSMinimumQoS . . . 73
8.53.2.8 pGPRSRequestedQos . . . 73
8.53.2.9 pImCnFlag . . . 73
8.53.2.10 pIPv4AddrPref . . . 73

8.53.2.11 pIPv6AddPref . . . 73
8.53.2.12 pPassword . . . 73
8.53.2.13 pPasswordSize . . . 73
8.53.2.14 pPcscfAddrUsingDhcp . . . 73
8.53.2.15 pPcscfAddrUsingPCO . . . 73
8.53.2.16 pPDNInactivTimeout . . . 73
8.53.2.17 pPdpAccessConFlag . . . 73
8.53.2.18 pPdpContext . . . 73
8.53.2.19 pPdpDataCompType . . . 73
8.53.2.20 pPdpHdrCompType . . . 73
8.53.2.21 pPDPtype . . . 73
8.53.2.22 pPriDNSIPv4AddPref . . . 73
8.53.2.23 pPriDNSIPv6addpref . . . 73
8.53.2.24 pPrimaryID . . . 74
8.53.2.25 pProfilename . . . 74
8.53.2.26 pProfilenameSize . . . 74
8.53.2.27 pQosClassID . . . 74
8.53.2.28 pSecDNSIPv4AddPref . . . 74
8.53.2.29 pSecDNSIPv6addpref . . . 74
8.53.2.30 pSecondaryFlag . . . 74
8.53.2.31 pTFTID1Params . . . 74
8.53.2.32 pTFTID2Params . . . 74
8.53.2.33 pUMTSMinQoS . . . 74
8.53.2.34 pUMTSMinQosSigInd . . . 74
8.53.2.35 pUMTSReqQoS . . . 74
8.53.2.36 pUMTSReqQoSSigInd . . . 74
8.53.2.37 pUsername . . . 74
8.53.2.38 pUsernameSize . . . 74
8.54 LibpackProfile3GPP2 Struct Reference . . . 74
8.54.1 Detailed Description . . . 75
8.54.2 Field Documentation . . . 78
8.54.2.1 pAllowLinger . . . 78
8.54.2.2 pAPNClass3GPP2 . . . 78
8.54.2.3 pAPNEnabled3GPP2 . . . 78
8.54.2.4 pApnString . . . 78
8.54.2.5 pApnStringSize . . . 78
8.54.2.6 pAppPriority . . . 78
8.54.2.7 pAppType . . . 78
8.54.2.8 pAuthPassword . . . 78
8.54.2.9 pAuthPasswordSize . . . 78

8.54.2.10 pAuthProtocol . . . 79
8.54.2.11 pAuthRetryCount . . . 79
8.54.2.12 pAuthTimeout . . . 79
8.54.2.13 pDataMode . . . 79
8.54.2.14 pDataRate . . . 79
8.54.2.15 pIpcpAckTimeout . . . 79
8.54.2.16 pIpcpCreqRetryCount . . . 79
8.54.2.17 pIsPcscfAddressNedded . . . 79
8.54.2.18 pLcpAckTimeout . . . 79
8.54.2.19 pLcpCreqRetryCount . . . 79
8.54.2.20 pNegoDnsSrvrPref . . . 79
8.54.2.21 pPDNInactivTimeout3GPP2 . . . 79
8.54.2.22 pPdnType . . . 79
8.54.2.23 pPppSessCloseTimer1x . . . 79
8.54.2.24 pPppSessCloseTimerDO . . . 79
8.54.2.25 pPrimaryV4DnsAddress . . . 79
8.54.2.26 pPriV6DnsAddress . . . 79
8.54.2.27 pRATType . . . 79
8.54.2.28 pSecondaryV4DnsAddress . . . 79
8.54.2.29 pSecV6DnsAddress . . . 79
8.54.2.30 pUserId . . . 79
8.54.2.31 pUserIdSize . . . 79
8.55 LibPackprofile_3GPP Struct Reference . . . 79
8.55.1 Detailed Description . . . 80
8.55.2 Field Documentation . . . 84
8.55.2.1 pAddrAllocPref . . . 84
8.55.2.2 pAPNClass . . . 84
8.55.2.3 pAPNDisabledFlag . . . 84
8.55.2.4 pAPNName . . . 84
8.55.2.5 pAPNnameSize . . . 84
8.55.2.6 pAuthenticationPref . . . 84
8.55.2.7 pGPRSMinimumQoS . . . 84
8.55.2.8 pGPRSRequestedQos . . . 84
8.55.2.9 pImCnFlag . . . 84
8.55.2.10 pIPv4AddrPref . . . 84
8.55.2.11 pIPv6AddPref . . . 84
8.55.2.12 pPassword . . . 84
8.55.2.13 pPasswordSize . . . 84
8.55.2.14 pPcscfAddrUsingDhcp . . . 84
8.55.2.15 pPcscfAddrUsingPCO . . . 84

8.55.2.16 pPDNInactivTimeout . . . 84
8.55.2.17 pPdpAccessConFlag . . . 84
8.55.2.18 pPdpContext . . . 84
8.55.2.19 pPdpDataCompType . . . 84
8.55.2.20 pPdpHdrCompType . . . 84
8.55.2.21 pPDPtype . . . 84
8.55.2.22 pPriDNSIPv4AddPref . . . 84
8.55.2.23 pPriDNSIPv6addpref . . . 84
8.55.2.24 pPrimaryID . . . 85
8.55.2.25 pProfilename . . . 85
8.55.2.26 pProfilenameSize . . . 85
8.55.2.27 pQosClassID . . . 85
8.55.2.28 pSecDNSIPv4AddPref . . . 85
8.55.2.29 pSecDNSIPv6addpref . . . 85
8.55.2.30 pSecondaryFlag . . . 85
8.55.2.31 pTFTID1Params . . . 85
8.55.2.32 pTFTID2Params . . . 85
8.55.2.33 pUMTSMinQoS . . . 85
8.55.2.34 pUMTSMinQosSigInd . . . 85
8.55.2.35 pUMTSReqQoS . . . 85
8.55.2.36 pUMTSReqQoSSigInd . . . 85
8.55.2.37 pUsername . . . 85
8.55.2.38 pUsernameSize . . . 85
8.56 LibPackprofile_3GPP2 Struct Reference . . . 85
8.56.1 Detailed Description . . . 86
8.56.2 Field Documentation . . . 89
8.56.2.1 pAllowLinger . . . 89
8.56.2.2 pAPNClass3GPP2 . . . 89
8.56.2.3 pAPNEnabled3GPP2 . . . 89
8.56.2.4 pApnString . . . 89
8.56.2.5 pApnStringSize . . . 89
8.56.2.6 pAppPriority . . . 89
8.56.2.7 pAppType . . . 89
8.56.2.8 pAuthPassword . . . 89
8.56.2.9 pAuthPassword_tSize . . . 89
8.56.2.10 pAuthProtocol . . . 90
8.56.2.11 pAuthRetryCount . . . 90
8.56.2.12 pAuthTimeout . . . 90
8.56.2.13 pDataMode . . . 90
8.56.2.14 pDataRate . . . 90

8.56.2.15 pIpcpAckTimeout . . . 90
8.56.2.16 pIpcpCreqRetryCount . . . 90
8.56.2.17 pIsPcscfAddressNedded . . . 90
8.56.2.18 pLcpAckTimeout . . . 90
8.56.2.19 pLcpCreqRetryCount . . . 90
8.56.2.20 pNegoDnsSrvrPref . . . 90
8.56.2.21 pPDNInactivTimeout3GPP2 . . . 90
8.56.2.22 pPdnType . . . 90
8.56.2.23 pPppSessCloseTimer1x . . . 90
8.56.2.24 pPppSessCloseTimerDO . . . 90
8.56.2.25 pPrimaryV4DnsAddress . . . 90
8.56.2.26 pPriV6DnsAddress . . . 90
8.56.2.27 pRATType . . . 90
8.56.2.28 pSecondaryV4DnsAddress . . . 90
8.56.2.29 pSecV6DnsAddress . . . 90
8.56.2.30 pUserId . . . 90
8.56.2.31 pUserIdSize . . . 90
8.57 LibPackQosClassID Struct Reference . . . 90
8.57.1 Detailed Description . . . 91
8.57.2 Field Documentation . . . 91
8.57.2.1 gDlBitRate . . . 91
8.57.2.2 gUlBitRate . . . 91
8.57.2.3 maxDlBitRate . . . 91
8.57.2.4 maxUlBitRate . . . 91
8.57.2.5 QCI . . . 91
8.58 LibPackTFTIDParams Struct Reference . . . 91
8.58.1 Detailed Description . . . 92
8.58.2 Field Documentation . . . 92
8.58.2.1 destPortRangeEnd . . . 93
8.58.2.2 destPortRangeStart . . . 93
8.58.2.3 eValid . . . 93
8.58.2.4 filterId . . . 93
8.58.2.5 flowLabel . . . 93
8.58.2.6 IPSECSPI . . . 93
8.58.2.7 ipVersion . . . 93
8.58.2.8 nextHeader . . . 93
8.58.2.9 pSourceIP . . . 93
8.58.2.10 sourceIPMask . . . 93
8.58.2.11 srcPortRangeEnd . . . 93
8.58.2.12 srcPortRangeStart . . . 93

8.58.2.13 tosMask . . . 93
8.59 LibPackUMTSQoS Struct Reference . . . 93
8.59.1 Detailed Description . . . 93
8.59.2 Field Documentation . . . 95
8.59.2.1 deliveryErrSDU . . . 95
8.59.2.2 grntDownlinkBitrate . . . 95
8.59.2.3 grntUplinkBitrate . . . 95
8.59.2.4 maxDownlinkBitrate . . . 95
8.59.2.5 maxSDUSize . . . 95
8.59.2.6 maxUplinkBitrate . . . 95
8.59.2.7 qosDeliveryOrder . . . 95
8.59.2.8 resBerRatio . . . 95
8.59.2.9 sduErrorRatio . . . 95
8.59.2.10 trafficClass . . . 95
8.59.2.11 trafficPriority . . . 95
8.59.2.12 transferDelay . . . 95
8.60 LibPackUMTSReqQoSSigInd Struct Reference . . . 95
8.60.1 Detailed Description . . . 96
8.60.2 Field Documentation . . . 96
8.60.2.1 SigInd . . . 96
8.60.2.2 UMTSReqQoS . . . 96
8.61 loc_BdsSV Struct Reference . . . 96
8.61.1 Detailed Description . . . 96
8.61.2 Field Documentation . . . 97
8.61.2.1 id . . . 97
8.61.2.2 mask . . . 97
8.62 loc_BdsSVInfo Struct Reference . . . 97
8.62.1 Detailed Description . . . 97
8.62.2 Field Documentation . . . 97
8.62.2.1 len . . . 97
8.62.2.2 pSV . . . 97
8.63 loc_CellDb Struct Reference . . . 97
8.63.1 Detailed Description . . . 97
8.63.2 Field Documentation . . . 98
8.63.2.1 mask . . . 98
8.64 loc_ClkInfo Struct Reference . . . 98
8.64.1 Detailed Description . . . 98
8.64.2 Field Documentation . . . 99
8.64.2.1 mask . . . 99
8.65 loc_GnssData Struct Reference . . . 99

8.65.1 Detailed Description . . . 100
8.65.2 Field Documentation . . . 100
8.65.2.1 mask . . . 101
8.66 loc_gpsTime Struct Reference . . . 101
8.66.1 Detailed Description . . . 101
8.66.2 Field Documentation . . . 101
8.66.2.1 gpsTimeOfWeekMs . . . 101
8.66.2.2 gpsWeek . . . 101
8.67 loc_IPv4Info Struct Reference . . . 101
8.67.1 Detailed Description . . . 101
8.67.2 Field Documentation . . . 102
8.67.2.1 address . . . 102
8.67.2.2 port . . . 102
8.67.2.3 TlvPresent . . . 102
8.68 loc_IPv6Info Struct Reference . . . 102
8.68.1 Detailed Description . . . 102
8.68.2 Field Documentation . . . 102
8.68.2.1 address . . . 102
8.68.2.2 port . . . 102
8.68.2.3 TlvPresent . . . 102
8.69 loc_LocApplicationInfo Struct Reference . . . 102
8.69.1 Detailed Description . . . 103
8.69.2 Field Documentation . . . 103
8.69.2.1 appNameLength . . . 103
8.69.2.2 appProviderLength . . . 103
8.69.2.3 appVersionLength . . . 103
8.69.2.4 appVersionValid . . . 103
8.69.2.5 pAppName . . . 103
8.69.2.6 pAppProvider . . . 103
8.69.2.7 pAppVersion . . . 103
8.70 loc_precisionDilution Struct Reference . . . 103
8.70.1 Detailed Description . . . 104
8.70.2 Field Documentation . . . 104
8.70.2.1 HDOP . . . 104
8.70.2.2 PDOP . . . 104
8.70.2.3 VDOP . . . 104
8.71 loc_satelliteInfo Struct Reference . . . 104
8.71.1 Detailed Description . . . 104
8.71.2 Field Documentation . . . 106
8.71.2.1 azimuth . . . 106

8.71.2.2 elevation . . . 106
8.71.2.3 gnssSvId . . . 106
8.71.2.4 healthStatus . . . 106
8.71.2.5 snr . . . 106
8.71.2.6 svInfoMask . . . 106
8.71.2.7 svListLen . . . 106
8.71.2.8 svStatus . . . 106
8.71.2.9 system . . . 106
8.71.2.10 validMask . . . 106
8.72 loc_sensorDataUsage Struct Reference . . . 106
8.72.1 Detailed Description . . . 106
8.72.2 Field Documentation . . . 107
8.72.2.1 aidingIndicatorMask . . . 107
8.72.2.2 usageMask . . . 107
8.73 loc_SV Struct Reference . . . 107
8.73.1 Detailed Description . . . 107
8.73.2 Field Documentation . . . 108
8.73.2.1 id . . . 108
8.73.2.2 mask . . . 108
8.73.2.3 system . . . 108
8.74 loc_SVInfo Struct Reference . . . 108
8.74.1 Detailed Description . . . 108
8.74.2 Field Documentation . . . 108
8.74.2.1 len . . . 108
8.74.2.2 pSV . . . 108
8.75 loc_svUsedforFix Struct Reference . . . 108
8.75.1 Detailed Description . . . 109
8.75.2 Field Documentation . . . 109
8.75.2.1 gnssSvUsedList . . . 109
8.75.2.2 gnssSvUsedList_len . . . 109
8.76 loc_urlAddr Struct Reference . . . 109
8.76.1 Detailed Description . . . 109
8.76.2 Field Documentation . . . 109
8.76.2.1 address . . . 109
8.76.2.2 TlvPresent . . . 109
8.77 lteSSInfo Struct Reference . . . 110
8.77.1 Detailed Description . . . 110
8.77.2 Field Documentation . . . 110
8.77.2.1 rsrp . . . 110
8.77.2.2 rsrq . . . 110

8.77.2.3 rssi . . . 110
8.77.2.4 snr . . . 110
8.78 messageModeTlv Struct Reference . . . 110
8.78.1 Detailed Description . . . 110
8.78.2 Field Documentation . . . 111
8.78.2.1 MessageModeInfo . . . 111
8.78.2.2 TlvPresent . . . 111
8.79 nas_acqOrderPref Struct Reference . . . 111
8.79.1 Detailed Description . . . 111
8.79.2 Field Documentation . . . 111
8.79.2.1 acqOrdeLen . . . 111
8.79.2.2 pAcqOrder . . . 111
8.80 nas_ActPilotPNElement Struct Reference . . . 111
8.80.1 Detailed Description . . . 112
8.80.2 Field Documentation . . . 112
8.80.2.1 ActSetPilotPN . . . 112
8.80.2.2 ActSetPilotPNStrength . . . 112
8.81 nas_AddCDMASysInfo Struct Reference . . . 112
8.81.1 Detailed Description . . . 112
8.81.2 Field Documentation . . . 112
8.81.2.1 geoSysIdx . . . 112
8.81.2.2 regPrd . . . 112
8.82 nas_AddSysInfo Struct Reference . . . 112
8.82.1 Detailed Description . . . 113
8.82.2 Field Documentation . . . 113
8.82.2.1 cellBroadcastCap . . . 113
8.82.2.2 geoSysIdx . . . 113
8.83 nas_CallBarringSysInfo Struct Reference . . . 113
8.83.1 Detailed Description . . . 113
8.83.2 Field Documentation . . . 114
8.83.2.1 csBarStatus . . . 114
8.83.2.2 psBarStatus . . . 114
8.84 nas_callBarStatus Struct Reference . . . 114
8.84.1 Detailed Description . . . 114
8.84.2 Field Documentation . . . 115
8.84.2.1 csBarStatus . . . 115
8.84.2.2 psBarStatus . . . 115
8.85 nas_CDMAChannel Struct Reference . . . 115
8.85.1 Detailed Description . . . 115
8.85.2 Field Documentation . . . 115

8.85.2.1 priChA . . . 115
8.85.2.2 priChB . . . 115
8.85.2.3 secChA . . . 115
8.85.2.4 secChB . . . 115
8.86 nas_CDMAECIOThresh Struct Reference . . . 115
8.86.1 Detailed Description . . . 116
8.86.2 Field Documentation . . . 116
8.86.2.1 CDMAECIOThreshListLen . . . 116
8.86.2.2 pCDMAECIOThreshList . . . 116
8.87 nas_CDMAInfo Struct Reference . . . 116
8.87.1 Detailed Description . . . 116
8.87.2 Field Documentation . . . 117
8.87.2.1 baseId . . . 117
8.87.2.2 baseLat . . . 117
8.87.2.3 baseLong . . . 117
8.87.2.4 nid . . . 117
8.87.2.5 refpn . . . 117
8.87.2.6 sid . . . 117
8.88 nas_CDMARSSIThresh Struct Reference . . . 117
8.88.1 Detailed Description . . . 117
8.88.2 Field Documentation . . . 117
8.88.2.1 CDMARSSIThreshListLen . . . 117
8.88.2.2 pCDMARSSIThreshList . . . 117
8.89 nas_CDMASysInfo Struct Reference . . . 118
8.89.1 Detailed Description . . . 118
8.89.2 Field Documentation . . . 120
8.89.2.1 baseId . . . 120
8.89.2.2 baseLat . . . 121
8.89.2.3 baseLong . . . 121
8.89.2.4 bsInfoValid . . . 121
8.89.2.5 bsPRev . . . 121
8.89.2.6 bsPRevValid . . . 121
8.89.2.7 ccsSupported . . . 121
8.89.2.8 ccsSupportedValid . . . 121
8.89.2.9 cdmaSysIdValid . . . 121
8.89.2.10 isSysPrlMatch . . . 121
8.89.2.11 isSysPrlMatchValid . . . 121
8.89.2.12 MCC . . . 121
8.89.2.13 MNC . . . 121
8.89.2.14 networkID . . . 121

8.89.2.15 networkIdValid . . . 121
8.89.2.16 packetZone . . . 121
8.89.2.17 packetZoneValid . . . 121
8.89.2.18 pRevInUse . . . 121
8.89.2.19 pRevInUseValid . . . 121
8.89.2.20 sysInfoCDMA . . . 121
8.89.2.21 systemID . . . 121
8.90 nas_CDMASysInfoExt Struct Reference . . . 121
8.90.1 Detailed Description . . . 121
8.90.2 Field Documentation . . . 122
8.90.2.1 imsi_11_12 . . . 122
8.90.2.2 MCC . . . 122
8.91 nas_cellParams Struct Reference . . . 122
8.91.1 Detailed Description . . . 122
8.91.2 Field Documentation . . . 123
8.91.2.1 pci . . . 123
8.91.2.2 rsrp . . . 123
8.91.2.3 rsrq . . . 123
8.91.2.4 rssi . . . 123
8.91.2.5 srxlev . . . 123
8.92 nas_CommInfo Struct Reference . . . 123
8.92.1 Detailed Description . . . 123
8.92.2 Field Documentation . . . 124
8.92.2.1 imsRegState . . . 124
8.92.2.2 modemMode . . . 124
8.92.2.3 psState . . . 124
8.92.2.4 systemMode . . . 124
8.92.2.5 temperature . . . 124
8.93 nas_CSGID Struct Reference . . . 124
8.93.1 Detailed Description . . . 125
8.93.2 Field Documentation . . . 125
8.93.2.1 id . . . 125
8.93.2.2 mcc . . . 125
8.93.2.3 mnc . . . 125
8.93.2.4 mncPcsDigits . . . 125
8.93.2.5 rat . . . 125
8.94 nas_currentPLMN Struct Reference . . . 125
8.94.1 Detailed Description . . . 125
8.94.2 Field Documentation . . . 126
8.94.2.1 MCC . . . 126

8.94.2.2 MNC . . . 126
8.94.2.3 netDescr . . . 126
8.94.2.4 netDescrLength . . . 126
8.95 nas_dataSrvCapabilities Struct Reference . . . 126
8.95.1 Detailed Description . . . 126
8.95.2 Field Documentation . . . 127
8.95.2.1 dataCapabilities . . . 127
8.95.2.2 dataCapabilitiesLen . . . 127
8.96 nas_DataStatusDetail Struct Reference . . . 127
8.96.1 Detailed Description . . . 127
8.96.2 Field Documentation . . . 129
8.96.2.1 IPAddress . . . 129
8.96.2.2 LastErrCode . . . 129
8.97 nas_detailSvcInfo Struct Reference . . . 129
8.97.1 Detailed Description . . . 129
8.97.2 Field Documentation . . . 130
8.97.2.1 hdrHybrid . . . 130
8.97.2.2 hdrSrvStatus . . . 130
8.97.2.3 isSysForbidden . . . 130
8.97.2.4 srvCapability . . . 130
8.97.2.5 srvStatus . . . 130
8.98 nas_DeviceConfigDetail Struct Reference . . . 130
8.98.1 Detailed Description . . . 130
8.98.2 Field Documentation . . . 131
8.98.2.1 Chipset . . . 131
8.98.2.2 HWVersion . . . 131
8.98.2.3 QLIC . . . 131
8.98.2.4 Technology . . . 131
8.99 nas_dirNum Struct Reference . . . 131
8.99.1 Detailed Description . . . 131
8.99.2 Field Documentation . . . 131
8.99.2.1 dirNum . . . 132
8.99.2.2 dirNumLen . . . 132
8.100 nas_DRCParams Struct Reference . . . 132
8.100.1 Detailed Description . . . 132
8.100.2 Field Documentation . . . 132
8.100.2.1 DRCCover . . . 132
8.100.2.2 DRCValue . . . 132
8.101 nas_ecioListElement Struct Reference . . . 132
8.101.1 Detailed Description . . . 132

8.101.2 Field Documentation . . . 133
8.101.2.1 ecio . . . 133
8.101.2.2 radioIf . . . 133
8.102 nas_ECIOThresh Struct Reference . . . 133
8.102.1 Detailed Description . . . 133
8.102.2 Field Documentation . . . 133
8.102.2.1 ECIOThresListLen . . . 133
8.102.2.2 pECIOThresList . . . 133
8.103 nas_errorRateListElement Struct Reference . . . 133
8.103.1 Detailed Description . . . 133
8.103.2 Field Documentation . . . 134
8.103.2.1 errorRate . . . 134
8.103.2.2 radioIf . . . 134
8.104 nas_GERANInfo Struct Reference . . . 134
8.104.1 Detailed Description . . . 135
8.104.2 Field Documentation . . . 136
8.104.2.1 arfcn . . . 136
8.104.2.2 bsic . . . 136
8.104.2.3 cellID . . . 136
8.104.2.4 insNmrCellInfo . . . 136
8.104.2.5 lac . . . 136
8.104.2.6 nmrInst . . . 136
8.104.2.7 plmn . . . 136
8.104.2.8 rxLev . . . 136
8.104.2.9 timingAdvance . . . 136
8.105 nas_geranInstInfo Struct Reference . . . 136
8.105.1 Detailed Description . . . 136
8.105.2 Field Documentation . . . 136
8.105.2.1 geranArfcn . . . 136
8.105.2.2 geranBsicBcc . . . 137
8.105.2.3 geranBsicNcc . . . 137
8.105.2.4 geranRssi . . . 137
8.106 nas_gsmCellInfo Struct Reference . . . 137
8.106.1 Detailed Description . . . 137
8.106.2 Field Documentation . . . 137
8.106.2.1 arfcn . . . 138
8.106.2.2 band1900 . . . 138
8.106.2.3 bsicId . . . 138
8.106.2.4 cellIdValid . . . 138
8.106.2.5 rssi . . . 138

8.106.2.6 srxlev . . . 138
8.107 nas_GSMRSSIThresh Struct Reference . . . 138
8.107.1 Detailed Description . . . 138
8.107.2 Field Documentation . . . 138
8.107.2.1 GSMRSSIThreshListLen . . . 138
8.107.2.2 pGSMRSSIThreshList . . . 138
8.108 nas_GSMSrvStatusInfo Struct Reference . . . 138
8.108.1 Detailed Description . . . 138
8.108.2 Field Documentation . . . 139
8.108.2.1 isPrefDataPath . . . 139
8.108.2.2 srvStatus . . . 139
8.108.2.3 trueSrvStatus . . . 139
8.109 nas_GSMSysInfo Struct Reference . . . 139
8.109.1 Detailed Description . . . 140
8.109.2 Field Documentation . . . 141
8.109.2.1 cellId . . . 141
8.109.2.2 cellIdValid . . . 141
8.109.2.3 dtmSupp . . . 141
8.109.2.4 dtmSuppValid . . . 142
8.109.2.5 egprsSupp . . . 142
8.109.2.6 egprsSuppValid . . . 142
8.109.2.7 lac . . . 142
8.109.2.8 lacValid . . . 142
8.109.2.9 MCC . . . 142
8.109.2.10 MNC . . . 142
8.109.2.11 networkIdValid . . . 142
8.109.2.12 regRejectInfoValid . . . 142
8.109.2.13 rejCause . . . 142
8.109.2.14 rejectSrvDomain . . . 142
8.109.2.15 sysInfoGSM . . . 142
8.110 nas_HDRECIOThresh Struct Reference . . . 142
8.110.1 Detailed Description . . . 142
8.110.2 Field Documentation . . . 142
8.110.2.1 HDRECIOThreshListLen . . . 142
8.110.2.2 pHDRECIOThreshList . . . 142
8.111 nas_HDRIOThresh Struct Reference . . . 142
8.111.1 Detailed Description . . . 143
8.111.2 Field Documentation . . . 143
8.111.2.1 HDRIOThreshListLen . . . 143
8.111.2.2 pHDRIOThreshList . . . 143

8.112nas_HDRRSSIThresh Struct Reference . . . 143
8.112.1 Detailed Description . . . 143
8.112.2 Field Documentation . . . 143
8.112.2.1 HDRRSSIThreshListLen . . . 143
8.112.2.2 pHDRRSSIThreshList . . . 144
8.113nas_HDRSINRThresh Struct Reference . . . 144
8.113.1 Detailed Description . . . 144
8.113.2 Field Documentation . . . 144
8.113.2.1 HDRSINRThresListLen . . . 144
8.113.2.2 pHDRSINRThresList . . . 144
8.114nas_HDRSINRThreshold Struct Reference . . . 144
8.114.1 Detailed Description . . . 144
8.114.2 Field Documentation . . . 145
8.114.2.1 HDRSINRThreshListLen . . . 145
8.114.2.2 pHDRSINRThreshList . . . 145
8.115nas_HDRSysInfo Struct Reference . . . 145
8.115.1 Detailed Description . . . 145
8.115.2 Field Documentation . . . 147
8.115.2.1 hdrActiveProt . . . 147
8.115.2.2 hdrActiveProtValid . . . 147
8.115.2.3 hdrPersonality . . . 147
8.115.2.4 hdrPersonalityValid . . . 147
8.115.2.5 is856SysId . . . 147
8.115.2.6 is856SysIdValid . . . 147
8.115.2.7 isSysPrlMatch . . . 147
8.115.2.8 isSysPrlMatchValid . . . 147
8.115.2.9 sysInfoHDR . . . 147
8.116nas_homeSIDNID Struct Reference . . . 147
8.116.1 Detailed Description . . . 147
8.116.2 Field Documentation . . . 147
8.116.2.1 numInstances . . . 147
8.116.2.2 SidNid . . . 147
8.117nas_infoInterFreq Struct Reference . . . 148
8.117.1 Detailed Description . . . 148
8.117.2 Field Documentation . . . 148
8.117.2.1 cell_resel_priority . . . 148
8.117.2.2 cellInterFreqParams . . . 148
8.117.2.3 cells_len . . . 148
8.117.2.4 earfcn . . . 149
8.117.2.5 threshXHigh . . . 149

8.117.2.6 threshXLow . . . 149
8.118 nas_IOThresh Struct Reference . . . 149
8.118.1 Detailed Description . . . 149
8.118.2 Field Documentation . . . 149
8.118.2.1 IOThresListLen . . . 149
8.118.2.2 pIOThresList . . . 149
8.119 nas_lteEARFCN Struct Reference . . . 149
8.119.1 Detailed Description . . . 149
8.119.2 Field Documentation . . . 150
8.119.2.1 earfcn0 . . . 150
8.119.2.2 earfcn1 . . . 150
8.119.2.3 status . . . 150
8.120 nas_lteGsmCellInfo Struct Reference . . . 150
8.120.1 Detailed Description . . . 150
8.120.2 Field Documentation . . . 151
8.120.2.1 cellReselPriority . . . 151
8.120.2.2 cells_len . . . 151
8.120.2.3 GsmCellInfo . . . 151
8.120.2.4 nccPermitted . . . 151
8.120.2.5 threshGsmHigh . . . 151
8.120.2.6 threshGsmLow . . . 151
8.121 nas_LTEInfo Struct Reference . . . 151
8.121.1 Detailed Description . . . 151
8.121.2 Field Documentation . . . 153
8.121.2.1 band . . . 153
8.121.2.2 bandwidth . . . 153
8.121.2.3 emmConnState . . . 153
8.121.2.4 emmState . . . 153
8.121.2.5 emmSubState . . . 153
8.121.2.6 RXChan . . . 153
8.121.2.7 TXChan . . . 153
8.122 nas_LTEInfoInterfreq Struct Reference . . . 154
8.122.1 Detailed Description . . . 154
8.122.2 Field Documentation . . . 154
8.122.2.1 freqsLen . . . 154
8.122.2.2 InfoInterfreq . . . 154
8.122.2.3 ueInIdle . . . 154
8.123 nas_LTEInfoIntrafreq Struct Reference . . . 154
8.123.1 Detailed Description . . . 155
8.123.2 Field Documentation . . . 156

8.123.2.1 CellParams . . . 156
8.123.2.2 cellReselPriority . . . 156
8.123.2.3 cellsLen . . . 156
8.123.2.4 earfcn . . . 156
8.123.2.5 globalCellId . . . 156
8.123.2.6 plmn . . . 156
8.123.2.7 servingCellId . . . 156
8.123.2.8 sIntraSearch . . . 156
8.123.2.9 sNonIntraSearch . . . 156
8.123.2.10 tac . . . 156
8.123.2.11 threshServingLow . . . 156
8.123.2.12 ueInIdle . . . 156
8.124 nas_LTEInfoNeighboringGSM Struct Reference . . . 156
8.124.1 Detailed Description . . . 157
8.124.2 Field Documentation . . . 157
8.124.2.1 freqsLen . . . 157
8.124.2.2 LteGsmCellInfo . . . 157
8.124.2.3 ueInIdle . . . 157
8.125 nas_LTEInfoNeighboringWCDMA Struct Reference . . . 157
8.125.1 Detailed Description . . . 157
8.125.2 Field Documentation . . . 158
8.125.2.1 freqsLen . . . 158
8.125.2.2 LTEWCDMACellInfo . . . 158
8.125.2.3 ueInIdle . . . 158
8.126 nas_ltePCI Struct Reference . . . 158
8.126.1 Detailed Description . . . 158
8.126.2 Field Documentation . . . 158
8.126.2.1 earfcn . . . 158
8.126.2.2 pci . . . 158
8.126.2.3 status . . . 158
8.127 nas_lteRsrpinformation Struct Reference . . . 158
8.127.1 Detailed Description . . . 158
8.127.2 Field Documentation . . . 159
8.127.2.1 rsrplevel . . . 159
8.128 nas_LTERSRPThresh Struct Reference . . . 159
8.128.1 Detailed Description . . . 159
8.128.2 Field Documentation . . . 159
8.128.2.1 LTERSRPThreshListLen . . . 159
8.128.2.2 pLTERSRPThreshList . . . 159
8.129 nas_LTERSRQThresh Struct Reference . . . 159

8.129.1 Detailed Description . . . 159
8.129.2 Field Documentation . . . 160
8.129.2.1 LTERSRQThreshListLen . . . 160
8.129.2.2 pLTERSRQThreshList . . . 160
8.130 nas_LTERSSIThresh Struct Reference . . . 160
8.130.1 Detailed Description . . . 160
8.130.2 Field Documentation . . . 160
8.130.2.1 LTERSSIThreshListLen . . . 160
8.130.2.2 pLTERSSIThreshList . . . 160
8.131 nas_LTESigRptCfg Struct Reference . . . 160
8.131.1 Detailed Description . . . 161
8.131.2 Field Documentation . . . 161
8.131.2.1 avgPeriod . . . 161
8.131.2.2 rptRate . . . 161
8.132 nas_LTESigRptConfig Struct Reference . . . 161
8.132.1 Detailed Description . . . 161
8.132.2 Field Documentation . . . 162
8.132.2.1 avgPeriod . . . 162
8.132.2.2 rptRate . . . 162
8.133 nas_lteSnrinformation Struct Reference . . . 162
8.133.1 Detailed Description . . . 162
8.133.2 Field Documentation . . . 162
8.133.2.1 snrlevel . . . 163
8.134 nas_LTESNRThresh Struct Reference . . . 163
8.134.1 Detailed Description . . . 163
8.134.2 Field Documentation . . . 163
8.134.2.1 LTESNRThresListLen . . . 163
8.134.2.2 pLTESNRThresList . . . 163
8.135 nas_LTESNRThreshold Struct Reference . . . 163
8.135.1 Detailed Description . . . 163
8.135.2 Field Documentation . . . 164
8.135.2.1 LTESNRThreshListLen . . . 164
8.135.2.2 pLTESNRThreshList . . . 164
8.136 nas_LTESysInfo Struct Reference . . . 164
8.136.1 Detailed Description . . . 164
8.136.2 Field Documentation . . . 166
8.136.2.1 cellId . . . 166
8.136.2.2 cellIdValid . . . 166
8.136.2.3 lac . . . 166
8.136.2.4 lacValid . . . 166

8.136.2.5 MCC . . . 166
8.136.2.6 MNC . . . 166
8.136.2.7 networkIdValid . . . 166
8.136.2.8 regRejectInfoValid . . . 166
8.136.2.9 rejCause . . . 166
8.136.2.10 rejectSrvDomain . . . 166
8.136.2.11 sysInfoLTE . . . 166
8.136.2.12 tac . . . 166
8.136.2.13 tacValid . . . 166
8.137 nas_lteWcdmaCellInfo Struct Reference . . . 166
8.137.1 Detailed Description . . . 167
8.137.2 Field Documentation . . . 167
8.137.2.1 cellReselPriority . . . 167
8.137.2.2 cellsLen . . . 167
8.137.2.3 threshXhigh . . . 167
8.137.2.4 threshXlow . . . 167
8.137.2.5 uarfcn . . . 167
8.137.2.6 WCDMACellInfo . . . 167
8.138 nas_minBasedIMSI Struct Reference . . . 167
8.138.1 Detailed Description . . . 168
8.138.2 Field Documentation . . . 168
8.138.2.1 imsiM1112 . . . 168
8.138.2.2 imsiMS1 . . . 168
8.138.2.3 imsiMS2 . . . 168
8.138.2.4 mccM . . . 168
8.139 nas_MNRInfo Struct Reference . . . 168
8.139.1 Detailed Description . . . 168
8.139.2 Field Documentation . . . 169
8.139.2.1 mcc . . . 169
8.139.2.2 mnc . . . 169
8.139.2.3 rat . . . 169
8.140 nas_namName Struct Reference . . . 169
8.140.1 Detailed Description . . . 169
8.140.2 Field Documentation . . . 169
8.140.2.1 namName . . . 169
8.140.2.2 namNameLen . . . 169
8.141 nas_netSelectionPref Struct Reference . . . 169
8.141.1 Detailed Description . . . 170
8.141.2 Field Documentation . . . 170
8.141.2.1 mcc . . . 170

8.141.2.2 mnc . . . 170
8.141.2.3 netReg . . . 170
8.142 nas_NetworkStat1x Struct Reference . . . 170
8.142.1 Detailed Description . . . 170
8.142.2 Field Documentation . . . 172
8.142.2.1 ActSetCnt . . . 172
8.142.2.2 NeighborSetCnt . . . 172
8.142.2.3 pActPilotPNElements . . . 172
8.142.2.4 pNeighborSetPilotPN . . . 172
8.142.2.5 RX_EC_IO . . . 172
8.142.2.6 RX_PWR . . . 172
8.142.2.7 SO . . . 172
8.142.2.8 State . . . 172
8.142.2.9 TX_PWR . . . 172
8.143 nas_NetworkStatEVDO Struct Reference . . . 172
8.143.1 Detailed Description . . . 172
8.143.2 Field Documentation . . . 173
8.143.2.1 MACIndex . . . 174
8.143.2.2 PER . . . 174
8.143.2.3 PilotEnergy . . . 174
8.143.2.4 pSectorID . . . 174
8.143.2.5 RX_PWR . . . 174
8.143.2.6 SectorIDLen . . . 174
8.143.2.7 SNR . . . 174
8.143.2.8 State . . . 174
8.144 nas_nmrCellInfo Struct Reference . . . 174
8.144.1 Detailed Description . . . 174
8.144.2 Field Documentation . . . 175
8.144.2.1 nmrArfcn . . . 175
8.144.2.2 nmrBsic . . . 175
8.144.2.3 nmrCellID . . . 175
8.144.2.4 nmrLac . . . 175
8.144.2.5 nmrPlmn . . . 175
8.144.2.6 nmrRxLev . . . 175
8.145 nas_operatorNameString Struct Reference . . . 175
8.145.1 Detailed Description . . . 175
8.145.2 Field Documentation . . . 175
8.145.2.1 PLMNName . . . 175
8.146 nas_OperatorPLMNData Struct Reference . . . 175
8.146.1 Detailed Description . . . 176

8.146.2 Field Documentation . . . 176
8.146.2.1 lac1 . . . 176
8.146.2.2 lac2 . . . 176
8.146.2.3 mcc . . . 176
8.146.2.4 mnc . . . 176
8.146.2.5 PLMNRecID . . . 176
8.147 nas_operatorPLMNList Struct Reference . . . 176
8.147.1 Detailed Description . . . 176
8.147.2 Field Documentation . . . 177
8.147.2.1 numInstance . . . 177
8.147.2.2 PLMNData . . . 177
8.148 nas_PhyCaAggPcellInfo Struct Reference . . . 177
8.148.1 Detailed Description . . . 177
8.148.2 Field Documentation . . . 177
8.148.2.1 dl_bw_value . . . 177
8.148.2.2 freq . . . 177
8.148.2.3 iLTEbandValue . . . 178
8.148.2.4 pci . . . 178
8.148.2.5 TlvPresent . . . 178
8.149 nas_PhyCaAggScellDlBw Struct Reference . . . 178
8.149.1 Detailed Description . . . 178
8.149.2 Field Documentation . . . 178
8.149.2.1 dl_bw_value . . . 178
8.149.2.2 TlvPresent . . . 178
8.150 nas_PhyCaAggScellIndex Struct Reference . . . 178
8.150.1 Detailed Description . . . 178
8.150.2 Field Documentation . . . 179
8.150.2.1 scell_idx . . . 179
8.150.2.2 TlvPresent . . . 179
8.151 nas_PhyCaAggScellIndType Struct Reference . . . 179
8.151.1 Detailed Description . . . 179
8.151.2 Field Documentation . . . 179
8.151.2.1 freq . . . 179
8.151.2.2 pci . . . 179
8.151.2.3 scell_state . . . 179
8.151.2.4 TlvPresent . . . 179
8.152 nas_PhyCaAggScellInfo Struct Reference . . . 179
8.152.1 Detailed Description . . . 180
8.152.2 Field Documentation . . . 182
8.152.2.1 dl_bw_value . . . 182

8.152.2.2 freq . . . 182
8.152.2.3 iLTEbandValue . . . 182
8.152.2.4 pci . . . 182
8.152.2.5 scell_state . . . 182
8.152.2.6 TlvPresent . . . 182
8.153 nas_PilotSetData Struct Reference . . . 182
8.153.1 Detailed Description . . . 182
8.153.2 Field Documentation . . . 182
8.153.2.1 NumPilots . . . 182
8.153.2.2 pPilotSetInfo . . . 182
8.154 nas_PilotSetParams Struct Reference . . . 182
8.154.1 Detailed Description . . . 183
8.154.2 Field Documentation . . . 183
8.154.2.1 PilotPN . . . 183
8.154.2.2 PilotStrength . . . 183
8.154.2.3 PilotType . . . 183
8.155 nas_PLMNNetworkName Struct Reference . . . 183
8.155.1 Detailed Description . . . 183
8.155.2 Field Documentation . . . 183
8.155.2.1 numInstance . . . 183
8.155.2.2 PLMNNetName . . . 183
8.156 nas_PLMNNetworkNameData Struct Reference . . . 184
8.156.1 Detailed Description . . . 184
8.156.2 Field Documentation . . . 185
8.156.2.1 codingScheme . . . 185
8.156.2.2 countryInitials . . . 185
8.156.2.3 longName . . . 185
8.156.2.4 longNameLen . . . 185
8.156.2.5 longNameSpareBits . . . 185
8.156.2.6 shortName . . . 185
8.156.2.7 shortNameLen . . . 185
8.156.2.8 shortNameSpareBits . . . 185
8.157 nas_protocolSubtypeElement Struct Reference . . . 185
8.157.1 Detailed Description . . . 185
8.157.2 Field Documentation . . . 186
8.157.2.1 AccessMac . . . 186
8.157.2.2 AuthProt . . . 186
8.157.2.3 ControlMac . . . 186
8.157.2.4 EncryptProt . . . 186
8.157.2.5 ForwardMac . . . 186

8.157.2.6 IdleState . . . 186
8.157.2.7 KeyExchange . . . 186
8.157.2.8 MultDisc . . . 187
8.157.2.9 PhysicalLayer . . . 187
8.157.2.10 ReverseMac . . . 187
8.157.2.11 SecProt . . . 187
8.157.2.12 VirtStream . . . 187
8.158 nas_qaQmi3Gpp2TimeZone Struct Reference . . . 187
8.158.1 Detailed Description . . . 187
8.158.2 Field Documentation . . . 187
8.158.2.1 daylightSavings . . . 187
8.158.2.2 leapSeconds . . . 187
8.158.2.3 localTimeOffset . . . 187
8.159 nas_QmiNas3GppNetworkInfo Struct Reference . . . 187
8.159.1 Detailed Description . . . 188
8.159.2 Field Documentation . . . 188
8.159.2.1 Desription . . . 188
8.159.2.2 Forbidden . . . 188
8.159.2.3 InUse . . . 188
8.159.2.4 MCC . . . 189
8.159.2.5 MNC . . . 189
8.159.2.6 Preferred . . . 189
8.159.2.7 Roaming . . . 189
8.160 nas_QmiNas3GppNetworkRAT Struct Reference . . . 189
8.160.1 Detailed Description . . . 189
8.160.2 Field Documentation . . . 189
8.160.2.1 MCC . . . 189
8.160.2.2 MNC . . . 189
8.160.2.3 RAT . . . 189
8.161 nas_QmisNasPcsDigit Struct Reference . . . 189
8.161.1 Detailed Description . . . 190
8.161.2 Field Documentation . . . 190
8.161.2.1 includes_pcs_digit . . . 190
8.161.2.2 MCC . . . 190
8.161.2.3 MNC . . . 190
8.162 nas_RejectReasonTlv Struct Reference . . . 190
8.162.1 Detailed Description . . . 190
8.162.2 Field Documentation . . . 190
8.162.2.1 rejectCause . . . 190
8.162.2.2 serviceDomain . . . 190

8.162.2.3 TlvPresent . . . 190
8.163 nas_RFInfoTlv Struct Reference . . . 190
8.163.1 Detailed Description . . . 191
8.163.2 Field Documentation . . . 191
8.163.2.1 activeBandClass . . . 191
8.163.2.2 activeChannel . . . 191
8.163.2.3 radioInterface . . . 191
8.163.2.4 radioInterfaceSize . . . 191
8.163.2.5 TlvPresent . . . 191
8.164 nas_roamIndList Struct Reference . . . 191
8.164.1 Detailed Description . . . 191
8.164.2 Field Documentation . . . 192
8.164.2.1 numInstances . . . 192
8.164.2.2 radioInterface . . . 192
8.164.2.3 roamIndicator . . . 192
8.165 nas_RSRPThresh Struct Reference . . . 192
8.165.1 Detailed Description . . . 192
8.165.2 Field Documentation . . . 193
8.165.2.1 pRSRPThresList . . . 193
8.165.2.2 RSRPThresListLen . . . 193
8.166 nas_rsrqInformation Struct Reference . . . 193
8.166.1 Detailed Description . . . 193
8.166.2 Field Documentation . . . 193
8.166.2.1 radioIf . . . 193
8.166.2.2 rsrq . . . 193
8.167 nas_RSRQThresh Struct Reference . . . 193
8.167.1 Detailed Description . . . 193
8.167.2 Field Documentation . . . 194
8.167.2.1 pRSRQThresList . . . 194
8.167.2.2 RSRQThresListLen . . . 194
8.168 nas_RSSIThresh Struct Reference . . . 194
8.168.1 Detailed Description . . . 194
8.168.2 Field Documentation . . . 194
8.168.2.1 pRSSIThresList . . . 194
8.168.2.2 RSSIThresListLen . . . 195
8.169 nas_rxInfo Struct Reference . . . 195
8.169.1 Detailed Description . . . 195
8.169.2 Field Documentation . . . 195
8.169.2.1 ecio . . . 195
8.169.2.2 isRadioTuned . . . 195

8.169.2.3 phase . . . 195
8.169.2.4 rscp . . . 195
8.169.2.5 rsrp . . . 196
8.169.2.6 rxPower . . . 196
8.170 nas_RxSigInfo Struct Reference . . . 196
8.170.1 Detailed Description . . . 196
8.170.2 Field Documentation . . . 196
8.170.2.1 isRadioTuned . . . 196
8.170.2.2 rsrp . . . 196
8.170.2.3 rxChainIndex . . . 197
8.170.2.4 rxPower . . . 197
8.171 nas_rxSignalStrengthListElement Struct Reference . . . 197
8.171.1 Detailed Description . . . 197
8.171.2 Field Documentation . . . 197
8.171.2.1 radioIf . . . 197
8.171.2.2 rxSignalStrength . . . 197
8.172 nas_SccRxInfo Struct Reference . . . 197
8.172.1 Detailed Description . . . 198
8.172.2 Field Documentation . . . 198
8.172.2.1 numInstances . . . 198
8.172.2.2 rsrq . . . 198
8.172.2.3 sigInfo . . . 198
8.172.2.4 snr . . . 198
8.172.2.5 TlvPresent . . . 198
8.173 nas_serviceProviderName Struct Reference . . . 198
8.173.1 Detailed Description . . . 198
8.173.2 Field Documentation . . . 199
8.173.2.1 displayCondition . . . 199
8.173.2.2 spn . . . 199
8.173.2.3 spnLength . . . 199
8.174 nas_servSystem Struct Reference . . . 199
8.174.1 Detailed Description . . . 199
8.174.2 Field Documentation . . . 200
8.174.2.1 csAttachState . . . 200
8.174.2.2 numRadioInterfaces . . . 200
8.174.2.3 psAttachState . . . 200
8.174.2.4 radioInterface . . . 200
8.174.2.5 regState . . . 200
8.174.2.6 selNetwork . . . 200
8.175 nas_sidNid Struct Reference . . . 201

8.175.1 Detailed Description . . . 201
8.175.2 Field Documentation . . . 201
8.175.2.1 nid . . . 201
8.175.2.2 sid . . . 201
8.176 nas_SignalStrengthTlv Struct Reference . . . 201
8.176.1 Detailed Description . . . 201
8.176.2 Field Documentation . . . 201
8.176.2.1 radioInterface . . . 201
8.176.2.2 signalStrength . . . 201
8.176.2.3 TlvPresent . . . 201
8.177 nas_SLQSSignalStrengthsIndReq Struct Reference . . . 201
8.177.1 Detailed Description . . . 202
8.177.2 Field Documentation . . . 203
8.177.2.1 ecioDelta . . . 203
8.177.2.2 ecioThresholdList . . . 203
8.177.2.3 ecioThresholdListLen . . . 203
8.177.2.4 ioDelta . . . 203
8.177.2.5 lteRsrpDelta . . . 203
8.177.2.6 lteSnrDelta . . . 203
8.177.2.7 rsrqDelta . . . 203
8.177.2.8 rxSignalStrengthDelta . . . 203
8.177.2.9 sinrDelta . . . 203
8.177.2.10 sinrThresholdList . . . 203
8.177.2.11 sinrThresholdListLen . . . 203
8.178 nas_SLQSSignalStrengthsInformation Struct Reference . . . 203
8.178.1 Detailed Description . . . 204
8.178.2 Field Documentation . . . 204
8.178.2.1 ecioInfo . . . 204
8.178.2.2 errorRateInfo . . . 204
8.178.2.3 io . . . 204
8.178.2.4 lteRsrpinfo . . . 204
8.178.2.5 lteSnrinfo . . . 204
8.178.2.6 rsrqInfo . . . 204
8.178.2.7 rxSignalStrengthInfo . . . 204
8.178.2.8 sinr . . . 204
8.179 nas_SLQSSignalStrengthsTlv Struct Reference . . . 204
8.179.1 Detailed Description . . . 204
8.179.2 Field Documentation . . . 204
8.179.2.1 sSLQSSignalStrengthsInfo . . . 204
8.179.2.2 TlvPresent . . . 204

8.180 nas_SrvStatusInfo Struct Reference . . . 205
8.180.1 Detailed Description . . . 205
8.180.2 Field Documentation . . . 205
8.180.2.1 isPrefDataPath . . . 205
8.180.2.2 srvStatus . . . 205
8.181 nas_sysInfoCommon Struct Reference . . . 205
8.181.1 Detailed Description . . . 205
8.181.2 Field Documentation . . . 207
8.181.2.1 isSysForbidden . . . 207
8.181.2.2 isSysForbiddenValid . . . 207
8.181.2.3 roamStatus . . . 207
8.181.2.4 roamStatusValid . . . 207
8.181.2.5 srvCapability . . . 207
8.181.2.6 srvCapabilityValid . . . 207
8.181.2.7 srvDomain . . . 207
8.181.2.8 srvDomainValid . . . 207
8.182 nas_TDSCDMAECIOThresh Struct Reference . . . 208
8.182.1 Detailed Description . . . 208
8.182.2 Field Documentation . . . 208
8.182.2.1 pTDSCDMAECIOThreshList . . . 208
8.182.2.2 TDSCDMAECIOThreshListLen . . . 208
8.183 nas_TDSCDMARSCPThresh Struct Reference . . . 208
8.183.1 Detailed Description . . . 208
8.183.2 Field Documentation . . . 208
8.183.2.1 pTDSCDMARSCPThreshList . . . 208
8.183.2.2 TDSCDMARSCPThreshListLen . . . 209
8.184 nas_TDSCDMARSSIThresh Struct Reference . . . 209
8.184.1 Detailed Description . . . 209
8.184.2 Field Documentation . . . 209
8.184.2.1 pTDSCDMARSSIThreshList . . . 209
8.184.2.2 TDSCDMARSSIThreshListLen . . . 209
8.185 nas_TDSCDMASINRCONFThresh Struct Reference . . . 209
8.185.1 Detailed Description . . . 209
8.185.2 Field Documentation . . . 209
8.185.2.1 pTDSCDMASINRCONFThreshList . . . 209
8.185.2.2 TDSCDMASINRCONFThreshListLen . . . 210
8.186 nas_TDSCDMASINRThresh Struct Reference . . . 210
8.186.1 Detailed Description . . . 210
8.186.2 Field Documentation . . . 210
8.186.2.1 pTDSCDMASINRThreshList . . . 210

8.186.2.2 TDSCDMASINRThreshListLen . . . 210
8.187 nas_timeInfo Struct Reference . . . 210
8.187.1 Detailed Description . . . 210
8.187.2 Field Documentation . . . 212
8.187.2.1 day . . . 212
8.187.2.2 dayLtSavingAdj . . . 212
8.187.2.3 dayOfWeek . . . 212
8.187.2.4 hour . . . 212
8.187.2.5 minute . . . 212
8.187.2.6 month . . . 212
8.187.2.7 radioInterface . . . 212
8.187.2.8 second . . . 212
8.187.2.9 timeZone . . . 212
8.187.2.10 TlvPresent . . . 212
8.187.2.11 year . . . 212
8.188 nas_trueIMSI Struct Reference . . . 212
8.188.1 Detailed Description . . . 212
8.188.2 Field Documentation . . . 213
8.188.2.1 imsiT1112 . . . 213
8.188.2.2 imsiTaddrNum . . . 213
8.188.2.3 imsiTS1 . . . 213
8.188.2.4 imsiTS2 . . . 213
8.188.2.5 mccT . . . 213
8.189 nas_txInfo Struct Reference . . . 213
8.189.1 Detailed Description . . . 213
8.189.2 Field Documentation . . . 213
8.189.2.1 isInTraffic . . . 213
8.189.2.2 txPower . . . 213
8.190 nas_UMTSInfo Struct Reference . . . 213
8.190.1 Detailed Description . . . 214
8.190.2 Field Documentation . . . 214
8.190.2.1 cellID . . . 215
8.190.2.2 ecio . . . 215
8.190.2.3 geranInst . . . 215
8.190.2.4 GeranInstInfo . . . 215
8.190.2.5 lac . . . 215
8.190.2.6 plmn . . . 215
8.190.2.7 psc . . . 215
8.190.2.8 rscp . . . 215
8.190.2.9 uarfcn . . . 215

8.190.2.10umtsInst . . . 215
8.190.2.11UMTSInstInfo . . . 215
8.191nas_UMTSinstInfo Struct Reference . . . 215
8.191.1 Detailed Description . . . 215
8.191.2 Field Documentation . . . 215
8.191.2.1 umtsEcio . . . 215
8.191.2.2 umtsPsc . . . 216
8.191.2.3 umtsRscp . . . 216
8.191.2.4 umtsUarfcn . . . 216
8.192nas_umtsLTENbrCell Struct Reference . . . 216
8.192.1 Detailed Description . . . 216
8.192.2 Field Documentation . . . 216
8.192.2.1 cellIsTDD . . . 216
8.192.2.2 earfcn . . . 216
8.192.2.3 pci . . . 216
8.192.2.4 rsrp . . . 217
8.192.2.5 rsrq . . . 217
8.192.2.6 srxlev . . . 217
8.193nas_UniversalTime Struct Reference . . . 217
8.193.1 Detailed Description . . . 217
8.193.2 Field Documentation . . . 218
8.193.2.1 day . . . 218
8.193.2.2 dayOfWeek . . . 218
8.193.2.3 hour . . . 218
8.193.2.4 minute . . . 218
8.193.2.5 month . . . 218
8.193.2.6 second . . . 218
8.193.2.7 year . . . 218
8.194nas_wcdmaCellInfo Struct Reference . . . 218
8.194.1 Detailed Description . . . 218
8.194.2 Field Documentation . . . 218
8.194.2.1 cpich_ecno . . . 218
8.194.2.2 cpich_rscp . . . 219
8.194.2.3 psc . . . 219
8.194.2.4 srxlev . . . 219
8.195nas_WCDMAECIOThresh Struct Reference . . . 219
8.195.1 Detailed Description . . . 219
8.195.2 Field Documentation . . . 219
8.195.2.1 pWCDMAECIOThreshList . . . 219
8.195.2.2 WCDMAECIOThreshListLen . . . 219

8.196 nas_WCDMAInfoLTENeighborCell Struct Reference . . . 219
8.196.1 Detailed Description . . . 219
8.196.2 Field Documentation . . . 220
8.196.2.1 UMTSLTENbrCell . . . 220
8.196.2.2 umtsLTENbrCellLen . . . 220
8.196.2.3 wcdmaRRCState . . . 220
8.197 nas_WCDMARSSIThresh Struct Reference . . . 220
8.197.1 Detailed Description . . . 220
8.197.2 Field Documentation . . . 221
8.197.2.1 pWCDMARSSIThreshList . . . 221
8.197.2.2 WCDMARSSIThreshListLen . . . 221
8.198 nas_WCDMASysInfo Struct Reference . . . 221
8.198.1 Detailed Description . . . 221
8.198.2 Field Documentation . . . 223
8.198.2.1 cellId . . . 223
8.198.2.2 cellIdValid . . . 223
8.198.2.3 hsCallStatus . . . 223
8.198.2.4 hsCallStatusValid . . . 224
8.198.2.5 hsInd . . . 224
8.198.2.6 hsIndValid . . . 224
8.198.2.7 lac . . . 224
8.198.2.8 lacValid . . . 224
8.198.2.9 MCC . . . 224
8.198.2.10 MNC . . . 224
8.198.2.11 networkIdValid . . . 224
8.198.2.12 psc . . . 224
8.198.2.13 pscValid . . . 224
8.198.2.14 regRejectInfoValid . . . 224
8.198.2.15 rejCause . . . 224
8.198.2.16 rejectSrvDomain . . . 224
8.198.2.17 sysInfoWCDMA . . . 224
8.199 nas_wcdmaUARFCN Struct Reference . . . 224
8.199.1 Detailed Description . . . 224
8.199.2 Field Documentation . . . 224
8.199.2.1 status . . . 224
8.199.2.2 uarfcn . . . 224
8.200 NASBandPreferenceTlv Struct Reference . . . 225
8.200.1 Detailed Description . . . 225
8.200.2 Field Documentation . . . 225
8.200.2.1 band_pref . . . 225

8.200.2.2 TlvPresent . . . 225
8.201 NASEmergencyModeTlv Struct Reference . . . 225
8.201.1 Detailed Description . . . 225
8.201.2 Field Documentation . . . 225
8.201.2.1 EmerMode . . . 225
8.201.2.2 TlvPresent . . . 225
8.202 NasGetLTECphyCaInfo Struct Reference . . . 225
8.202.1 Field Documentation . . . 226
8.202.1.1 PhyCaAggPcellInfo . . . 226
8.202.1.2 PhyCaAggScellDlBw . . . 226
8.202.1.3 PhyCaAggScellIndex . . . 226
8.202.1.4 PhyCaAggScellIndType . . . 226
8.202.1.5 PhyCaAggScellInfo . . . 226
8.203 NASGWAcqOrderPrefTlv Struct Reference . . . 226
8.203.1 Detailed Description . . . 226
8.203.2 Field Documentation . . . 226
8.203.2.1 GWAcqOrderPref . . . 226
8.203.2.2 TlvPresent . . . 226
8.204 NASLTEBandPreferenceTlv Struct Reference . . . 226
8.204.1 Detailed Description . . . 226
8.204.2 Field Documentation . . . 227
8.204.2.1 LTEBandPref . . . 227
8.204.2.2 TlvPresent . . . 227
8.205 NASLteNasReleaseInfoTlv Struct Reference . . . 227
8.205.1 Detailed Description . . . 227
8.205.2 Field Documentation . . . 227
8.205.2.1 nas_major . . . 227
8.205.2.2 nas_minor . . . 227
8.205.2.3 nas_release . . . 227
8.205.2.4 TlvPresent . . . 227
8.206 NASModePreferenceTlv Struct Reference . . . 227
8.206.1 Detailed Description . . . 228
8.206.2 Field Documentation . . . 228
8.206.2.1 ModePref . . . 228
8.206.2.2 TlvPresent . . . 228
8.207 NASNetSelPreferenceTlv Struct Reference . . . 228
8.207.1 Detailed Description . . . 228
8.207.2 Field Documentation . . . 228
8.207.2.1 NetSelPref . . . 228
8.207.2.2 TlvPresent . . . 228

8.208 NASOTAMessageTlv Struct Reference . . . 228
8.208.1 Detailed Description . . . 228
8.208.2 Field Documentation . . . 229
8.208.2.1 data_buf . . . 229
8.208.2.2 data_len . . . 229
8.208.2.3 message_type . . . 229
8.208.2.4 TlvPresent . . . 229
8.209 NASPhyCaAggPcellInfo Struct Reference . . . 229
8.209.1 Detailed Description . . . 229
8.209.2 Field Documentation . . . 230
8.209.2.1 dl_bw_value . . . 230
8.209.2.2 freq . . . 230
8.209.2.3 iLTEbandValue . . . 230
8.209.2.4 pci . . . 230
8.209.2.5 TlvPresent . . . 230
8.210 NASPhyCaAggScellDlBw Struct Reference . . . 230
8.210.1 Detailed Description . . . 230
8.210.2 Field Documentation . . . 230
8.210.2.1 dl_bw_value . . . 230
8.210.2.2 TlvPresent . . . 230
8.211 NASPhyCaAggScellIndex Struct Reference . . . 230
8.211.1 Detailed Description . . . 231
8.211.2 Field Documentation . . . 231
8.211.2.1 scell_idx . . . 231
8.211.2.2 TlvPresent . . . 231
8.212 NASPhyCaAggScellIndType Struct Reference . . . 231
8.212.1 Detailed Description . . . 231
8.212.2 Field Documentation . . . 232
8.212.2.1 freq . . . 232
8.212.2.2 pci . . . 232
8.212.2.3 scell_state . . . 232
8.212.2.4 TlvPresent . . . 232
8.213 NASPhyCaAggScellInfo Struct Reference . . . 232
8.213.1 Detailed Description . . . 232
8.213.2 Field Documentation . . . 233
8.213.2.1 dl_bw_value . . . 233
8.213.2.2 freq . . . 233
8.213.2.3 iLTEbandValue . . . 233
8.213.2.4 pci . . . 233
8.213.2.5 scell_state . . . 233

8.213.2.6 TlvPresent . . . 233
8.214 NASPRLPreferenceTlv Struct Reference . . . 233
8.214.1 Detailed Description . . . 233
8.214.2 Field Documentation . . . 233
8.214.2.1 PRLPref . . . 233
8.214.2.2 TlvPresent . . . 233
8.215 NASQmiCbkNasSwiOTAMessageInd Struct Reference . . . 233
8.215.1 Detailed Description . . . 233
8.215.2 Field Documentation . . . 234
8.215.2.1 nasRelInfoTlv . . . 234
8.215.2.2 otaMsgTlv . . . 234
8.215.2.3 timeTlv . . . 234
8.216 NASQmiCbkNasSystemSelPrefInd Struct Reference . . . 234
8.216.1 Detailed Description . . . 234
8.216.2 Field Documentation . . . 235
8.216.2.1 BPTlv . . . 235
8.216.2.2 EMTlv . . . 235
8.216.2.3 GWAOPTlv . . . 235
8.216.2.4 LBPTlv . . . 235
8.216.2.5 MPTlv . . . 235
8.216.2.6 NSPTlv . . . 235
8.216.2.7 PRLPTlv . . . 235
8.216.2.8 RPTlv . . . 235
8.216.2.9 SDPTlv . . . 235
8.217 NASRoamPreferenceTlv Struct Reference . . . 235
8.217.1 Detailed Description . . . 235
8.217.2 Field Documentation . . . 235
8.217.2.1 RoamPref . . . 236
8.217.2.2 TlvPresent . . . 236
8.218 NASServDomainPrefTlv Struct Reference . . . 236
8.218.1 Detailed Description . . . 236
8.218.2 Field Documentation . . . 236
8.218.2.1 SrvDomainPref . . . 236
8.218.2.2 TlvPresent . . . 236
8.219 NASServingSystemInfo Struct Reference . . . 236
8.219.1 Detailed Description . . . 236
8.219.2 Field Documentation . . . 237
8.219.2.1 csAttachState . . . 237
8.219.2.2 hdrPersonality . . . 237
8.219.2.3 psAttachState . . . 237

8.219.2.4 radioInterfaceList . . . 237
8.219.2.5 radioInterfaceNo . . . 237
8.219.2.6 registrationState . . . 237
8.219.2.7 selectedNetwork . . . 237
8.220 NASTimeInfoTlv Struct Reference . . . 237
8.220.1 Detailed Description . . . 238
8.220.2 Field Documentation . . . 238
8.220.2.1 time . . . 238
8.220.2.2 TlvPresent . . . 238
8.221 newMTMessageTlv Struct Reference . . . 238
8.221.1 Detailed Description . . . 238
8.221.2 Field Documentation . . . 238
8.221.2.1 MTMessageInfo . . . 238
8.221.2.2 TlvPresent . . . 238
8.222 pack_cat_CATSendEnvelopeCommand_t Struct Reference . . . 238
8.222.1 Detailed Description . . . 239
8.222.2 Field Documentation . . . 239
8.222.2.1 cmdID . . . 239
8.222.2.2 dataLen . . . 239
8.222.2.3 pData . . . 239
8.223 pack_cat_CATSendTerminalResponse_t Struct Reference . . . 239
8.223.1 Detailed Description . . . 239
8.223.2 Field Documentation . . . 240
8.223.2.1 dataLen . . . 240
8.223.2.2 pData . . . 240
8.223.2.3 refID . . . 240
8.224 pack_cat_SetCATEventCallback_t Struct Reference . . . 240
8.224.1 Detailed Description . . . 240
8.224.2 Field Documentation . . . 240
8.224.2.1 eventMask . . . 240
8.225 pack_dms_ActivateAutomatic_t Struct Reference . . . 241
8.225.1 Detailed Description . . . 241
8.225.2 Field Documentation . . . 241
8.225.2.1 actCode . . . 241
8.226 pack_dms_GetCustFeaturesV2_t Struct Reference . . . 241
8.226.1 Detailed Description . . . 241
8.226.2 Field Documentation . . . 241
8.226.2.1 cust_id . . . 241
8.226.2.2 list_type . . . 241
8.226.2.3 Tlvresult . . . 242

8.227 pack_dms_ResetToFactoryDefaults_t Struct Reference . . . 242
8.227.1 Detailed Description . . . 242
8.227.2 Field Documentation . . . 242
8.227.2.1 spc . . . 242
8.228 pack_dms_SetActivationStatusCallback_t Struct Reference . . . 242
8.228.1 Detailed Description . . . 242
8.228.2 Field Documentation . . . 242
8.228.2.1 activationState . . . 242
8.229 pack_dms_SetCrashAction_t Struct Reference . . . 242
8.229.1 Detailed Description . . . 243
8.229.2 Field Documentation . . . 243
8.229.2.1 crashAction . . . 243
8.230 pack_dms_SetCustFeature_t Struct Reference . . . 243
8.230.1 Detailed Description . . . 243
8.230.2 Field Documentation . . . 245
8.230.2.1 DHCPRelayEnabled . . . 245
8.230.2.2 DisableIMSI . . . 245
8.230.2.3 GpsEnable . . . 245
8.230.2.4 GPSLPM . . . 245
8.230.2.5 GPSSel . . . 245
8.230.2.6 IPFamSupport . . . 245
8.230.2.7 IsVoiceEnabled . . . 245
8.230.2.8 RMAutoConnect . . . 245
8.230.2.9 SMSSupport . . . 245
8.231 pack_dms_SetCustFeaturesV2_t Struct Reference . . . 245
8.231.1 Detailed Description . . . 245
8.231.2 Field Documentation . . . 246
8.231.2.1 cust_id . . . 246
8.231.2.2 cust_value . . . 246
8.231.2.3 Tlvresult . . . 246
8.231.2.4 value_length . . . 246
8.232 pack_dms_SetEventReport_t Struct Reference . . . 246
8.232.1 Detailed Description . . . 246
8.232.2 Field Documentation . . . 246
8.232.2.1 mode . . . 246
8.233 pack_dms_SetPower_t Struct Reference . . . 246
8.233.1 Detailed Description . . . 246
8.233.2 Field Documentation . . . 247
8.233.2.1 mode . . . 247
8.233.2.2 Tlvresult . . . 247

8.234 pack_dms_SetUSBComp_t Struct Reference . . . 247
8.234.1 Detailed Description . . . 247
8.234.2 Field Documentation . . . 248
8.234.2.1 TlvResult . . . 248
8.234.2.2 USBComp . . . 248
8.235 pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 248
8.235.1 Detailed Description . . . 248
8.235.2 Field Documentation . . . 248
8.235.2.1 resetInfoInd . . . 248
8.236 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 248
8.236.1 Detailed Description . . . 248
8.236.2 Field Documentation . . . 248
8.236.2.1 clear . . . 249
8.237 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 249
8.237.1 Detailed Description . . . 249
8.237.2 Field Documentation . . . 249
8.237.2.1 pDestSMSContent . . . 249
8.237.2.2 pDestSMSNum . . . 249
8.238 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 249
8.238.1 Detailed Description . . . 249
8.238.2 Field Documentation . . . 250
8.238.2.1 manString . . . 250
8.238.2.2 modelString . . . 250
8.238.2.3 plasmaIDString . . . 250
8.238.2.4 swVerString . . . 250
8.239 pack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 250
8.239.1 Detailed Description . . . 250
8.239.2 Field Documentation . . . 250
8.239.2.1 nameString . . . 250
8.239.2.2 versionString . . . 250
8.240 pack_dms_UIMChangePIN_t Struct Reference . . . 250
8.240.1 Detailed Description . . . 251
8.240.2 Field Documentation . . . 251
8.240.2.1 id . . . 251
8.240.2.2 newValue . . . 251
8.240.2.3 oldValue . . . 251
8.241 pack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 251
8.241.1 Detailed Description . . . 251
8.241.2 Field Documentation . . . 251
8.241.2.1 facility . . . 251

8.242pack_dms_UIMGetICCID_t Struct Reference . . . 252
8.242.1 Detailed Description . . . 252
8.242.2 Field Documentation . . . 252
8.242.2.1 TlvResult . . . 252
8.243pack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 252
8.243.1 Detailed Description . . . 252
8.243.2 Field Documentation . . . 253
8.243.2.1 facility . . . 253
8.243.2.2 facilityCk . . . 253
8.243.2.3 facilityState . . . 253
8.244pack_dms_UIMSetPINProtection_t Struct Reference . . . 253
8.244.1 Detailed Description . . . 253
8.244.2 Field Documentation . . . 253
8.244.2.1 bEnable . . . 253
8.244.2.2 id . . . 253
8.244.2.3 value . . . 253
8.245pack_dms_UIMUnblockControlKey_t Struct Reference . . . 253
8.245.1 Detailed Description . . . 253
8.245.2 Field Documentation . . . 254
8.245.2.1 facility . . . 254
8.245.2.2 facilityCk . . . 254
8.246pack_dms_UIMUnblockPIN_t Struct Reference . . . 254
8.246.1 Detailed Description . . . 254
8.246.2 Field Documentation . . . 254
8.246.2.1 id . . . 254
8.246.2.2 newPin . . . 254
8.246.2.3 pukValue . . . 254
8.247pack_dms_UIMVerifyPIN_t Struct Reference . . . 255
8.247.1 Detailed Description . . . 255
8.247.2 Field Documentation . . . 255
8.247.2.1 id . . . 255
8.247.2.2 value . . . 255
8.248pack_fms_GetImagesPreference_t Struct Reference . . . 255
8.248.1 Detailed Description . . . 255
8.248.2 Field Documentation . . . 255
8.248.2.1 TlvResult . . . 255
8.249pack_fms_GetStoredImages_t Struct Reference . . . 255
8.249.1 Detailed Description . . . 256
8.249.2 Field Documentation . . . 256
8.249.2.1 TlvResult . . . 256

8.250 pack_fms_SetImagesPreference_t Struct Reference . . . 256
8.250.1 Detailed Description . . . 256
8.250.2 Field Documentation . . . 257
8.250.2.1 bForceDownload . . . 257
8.250.2.2 imageListSize . . . 257
8.250.2.3 modemindex . . . 257
8.250.2.4 pImageList . . . 257
8.250.2.5 Tlvresult . . . 257
8.251 pack_loc_Delete_Assist_Data_t Struct Reference . . . 257
8.251.1 Detailed Description . . . 257
8.251.2 Field Documentation . . . 257
8.251.2.1 pBdsSVInfo . . . 257
8.251.2.2 pCellDb . . . 257
8.251.2.3 pClkInfo . . . 258
8.251.2.4 pGnssData . . . 258
8.251.2.5 pSVInfo . . . 258
8.251.2.6 Tlvresult . . . 258
8.252 pack_loc_EventRegister_t Struct Reference . . . 258
8.252.1 Detailed Description . . . 258
8.252.2 Field Documentation . . . 260
8.252.2.1 eventRegister . . . 260
8.252.2.2 Tlvresult . . . 260
8.253 pack_loc_SetExtPowerState_t Struct Reference . . . 260
8.253.1 Detailed Description . . . 260
8.253.2 Field Documentation . . . 260
8.253.2.1 extPowerState . . . 260
8.253.2.2 Tlvresult . . . 260
8.254 pack_loc_SetOperationMode_t Struct Reference . . . 260
8.254.1 Detailed Description . . . 261
8.254.2 Field Documentation . . . 261
8.254.2.1 mode . . . 261
8.254.2.2 Tlvresult . . . 261
8.255 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 261
8.255.1 Detailed Description . . . 261
8.255.2 Field Documentation . . . 261
8.255.2.1 Tlvresult . . . 261
8.255.2.2 xid . . . 262
8.256 pack_loc_SLQSLOCGetServer_t Struct Reference . . . 262
8.256.1 Detailed Description . . . 262
8.256.2 Field Documentation . . . 262

8.256.2.1 has_serverAddrTypeMask . . . 262
8.256.2.2 serverAddrTypeMask . . . 262
8.256.2.3 serverType . . . 262
8.257 pack_loc_SLQSLOCInjectPosition_t Struct Reference . . . 262
8.257.1 Detailed Description . . . 263
8.257.2 Field Documentation . . . 266
8.257.2.1 altitudeSrcInfo . . . 266
8.257.2.2 altitudeWrtEllipsoid . . . 266
8.257.2.3 altitudeWrtMeanSeaLevel . . . 266
8.257.2.4 has_altitudeSrcInfo . . . 266
8.257.2.5 has_altitudeWrtEllipsoid . . . 266
8.257.2.6 has_altitudeWrtMeanSeaLevel . . . 266
8.257.2.7 has_horConfidence . . . 266
8.257.2.8 has_horReliability . . . 266
8.257.2.9 has_horUncCircular . . . 266
8.257.2.10 has_latitude . . . 266
8.257.2.11 has_longitude . . . 266
8.257.2.12 has_positionSrc . . . 266
8.257.2.13 has_rawHorConfidence . . . 266
8.257.2.14 has_rawHorUncCircular . . . 266
8.257.2.15 has_timestampAge . . . 266
8.257.2.16 has_timestampUtc . . . 266
8.257.2.17 has_vertConfidence . . . 266
8.257.2.18 has_vertRelicability . . . 267
8.257.2.19 has_vertUnc . . . 267
8.257.2.20 horConfidence . . . 267
8.257.2.21 horReliability . . . 267
8.257.2.22 horUncCircular . . . 267
8.257.2.23 latitude . . . 267
8.257.2.24 longitude . . . 267
8.257.2.25 positionSrc . . . 267
8.257.2.26 rawHorConfidence . . . 267
8.257.2.27 rawHorUncCircular . . . 267
8.257.2.28 timestampAge . . . 267
8.257.2.29 timestampUtc . . . 267
8.257.2.30 vertConfidence . . . 267
8.257.2.31 vertReliability . . . 267
8.257.2.32 vertUnc . . . 267
8.258 pack_loc_SLQSLOCInjectSensorData_t Struct Reference . . . 267
8.258.1 Detailed Description . . . 267

8.258.2 Field Documentation . . . 268
8.258.2.1 acceleroData . . . 268
8.258.2.2 acceleroTimeSrc . . . 268
8.258.2.3 accelTemp . . . 268
8.258.2.4 gyroData . . . 268
8.258.2.5 gyroTemp . . . 269
8.258.2.6 gyroTimeSrc . . . 269
8.258.2.7 has_acceleroTimeSrc . . . 269
8.258.2.8 has_accelTemp . . . 269
8.258.2.9 has_accleroData . . . 269
8.258.2.10 has_gyroData . . . 269
8.258.2.11 has_gyroTemp . . . 269
8.258.2.12 has_gyroTimeSrc . . . 269
8.258.2.13 has_opaqueId . . . 269
8.258.2.14 opaqueId . . . 269
8.259 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference . . . 269
8.259.1 Detailed Description . . . 269
8.259.2 Field Documentation . . . 269
8.259.2.1 timeMsec . . . 269
8.259.2.2 timeUncMsec . . . 269
8.260 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference . . . 269
8.260.1 Detailed Description . . . 270
8.260.2 Field Documentation . . . 270
8.260.2.1 confidence . . . 270
8.260.2.2 has_confidence . . . 270
8.260.2.3 state . . . 270
8.261 pack_loc_Start_t Struct Reference . . . 270
8.261.1 Detailed Description . . . 270
8.261.2 Field Documentation . . . 271
8.261.2.1 pApplicationInfo . . . 271
8.261.2.2 pConfigAltitudeAssumed . . . 272
8.261.2.3 pHorizontalAccuracyLvl . . . 272
8.261.2.4 pIntermediateReportState . . . 272
8.261.2.5 pMinIntervalTime . . . 272
8.261.2.6 pRecurrenceType . . . 272
8.261.2.7 SessionId . . . 272
8.261.2.8 Tlvresult . . . 272
8.262 pack_loc_Stop_t Struct Reference . . . 272
8.262.1 Detailed Description . . . 272
8.262.2 Field Documentation . . . 272

8.262.2.1 SessionId . . . 272
8.262.2.2 Tlvresult . . . 272
8.263pack_nas_InitiateDomainAttach_t Struct Reference . . . 272
8.263.1 Detailed Description . . . 272
8.263.2 Field Documentation . . . 273
8.263.2.1 action . . . 273
8.264pack_nas_SetACCOLC_t Struct Reference . . . 273
8.264.1 Detailed Description . . . 273
8.264.2 Field Documentation . . . 273
8.264.2.1 accolc . . . 273
8.264.2.2 spc . . . 273
8.265pack_nas_SetCDMANetworkParameters_t Struct Reference . . . 273
8.265.1 Detailed Description . . . 273
8.265.2 Field Documentation . . . 274
8.265.2.1 pApplication . . . 274
8.265.2.2 pBroadcast . . . 274
8.265.2.3 pCustomSCP . . . 275
8.265.2.4 pForceRev0 . . . 275
8.265.2.5 pProtocol . . . 275
8.265.2.6 pRoaming . . . 275
8.265.2.7 pSPC . . . 275
8.266pack_nas_SetNetworkPreference_t Struct Reference . . . 275
8.266.1 Detailed Description . . . 275
8.266.2 Field Documentation . . . 275
8.266.2.1 Duration . . . 275
8.266.2.2 TechnologyPref . . . 275
8.266.2.3 Tlvresult . . . 276
8.267pack_nas_SLQSConfigSigInfo_t Struct Reference . . . 276
8.267.1 Detailed Description . . . 276
8.267.2 Field Documentation . . . 277
8.267.2.1 pECIOThresh . . . 277
8.267.2.2 pHDRSINRThresh . . . 277
8.267.2.3 pIOThresh . . . 277
8.267.2.4 pLTESigRptCfg . . . 277
8.267.2.5 pLTESNRThresh . . . 277
8.267.2.6 pRSRPThresh . . . 277
8.267.2.7 pRSRQThresh . . . 277
8.267.2.8 pRSSIThresh . . . 277
8.267.2.9 pTDSCDMASINRCONFThresh . . . 277
8.268pack_nas_SLQSGetPLMNName_t Struct Reference . . . 277

8.268.1 Detailed Description . . . 277
8.268.2 Field Documentation . . . 278
8.268.2.1 mcc . . . 278
8.268.2.2 mnc . . . 278
8.268.2.3 pMncPcsStatus . . . 278
8.269 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference . . . 278
8.269.1 Detailed Description . . . 278
8.269.2 Field Documentation . . . 278
8.269.2.1 pChangeDuration . . . 279
8.269.2.2 pMncPcsDigitStatus . . . 279
8.269.2.3 pMNRInfo . . . 279
8.269.2.4 regAction . . . 279
8.270 pack_nas_SLQSNasConfigSigInfo2_t Struct Reference . . . 279
8.270.1 Detailed Description . . . 280
8.270.2 Field Documentation . . . 282
8.270.2.1 pCDMAECIODelta . . . 282
8.270.2.2 pCDMAECIOThresh . . . 282
8.270.2.3 pCDMARSSIDelta . . . 282
8.270.2.4 pCDMARSSIThresh . . . 282
8.270.2.5 pGSMRSSIDelta . . . 282
8.270.2.6 pGSMRSSIThresh . . . 282
8.270.2.7 pHDRECIODelta . . . 282
8.270.2.8 pHDRECIOThresh . . . 282
8.270.2.9 pHDRIODelta . . . 282
8.270.2.10 pHDRIOThresh . . . 282
8.270.2.11 pHDRRSSIDelta . . . 282
8.270.2.12 pHDRRSSIThresh . . . 282
8.270.2.13 pHDRSINRDelta . . . 282
8.270.2.14 pHDRSINRThresh . . . 282
8.270.2.15 pLTERSRPDelta . . . 283
8.270.2.16 pLTERSRPThresh . . . 283
8.270.2.17 pLTERSRQDelta . . . 283
8.270.2.18 pLTERSRQThresh . . . 283
8.270.2.19 pLTERSSIDelta . . . 283
8.270.2.20 pLTERSSIThresh . . . 283
8.270.2.21 pLTESigRptConfig . . . 283
8.270.2.22 pLTESNRDelta . . . 283
8.270.2.23 pLTESNRThresh . . . 283
8.270.2.24 pTDSCDMAECIODelta . . . 283
8.270.2.25 pTDSCDMAECIOThresh . . . 283

8.270.2.26 pTDSCDMARSCPDelta . . . 283
8.270.2.27 pTDSCDMARSCPThresh . . . 283
8.270.2.28 pTDSCDMARSSIDelta . . . 283
8.270.2.29 pTDSCDMARSSIThresh . . . 283
8.270.2.30 pTDSCDMASINRDelta . . . 283
8.270.2.31 pTDSCDMASINRThresh . . . 283
8.270.2.32 pWCDMAECIODelta . . . 283
8.270.2.33 pWCDMAECIOThresh . . . 283
8.270.2.34 pWCDMARSSIDelta . . . 283
8.270.2.35 pWCDMARSSIThresh . . . 283
8.271 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 283
8.271.1 Detailed Description . . . 283
8.271.2 Field Documentation . . . 284
8.271.2.1 namID . . . 284
8.272 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 284
8.272.1 Detailed Description . . . 284
8.272.2 Field Documentation . . . 284
8.272.2.1 radio_if . . . 284
8.273 pack_nas_SLQSNasIndicationRegisterExt_t Struct Reference . . . 284
8.273.1 Detailed Description . . . 285
8.273.2 Field Documentation . . . 286
8.273.2.1 pDDTMInd . . . 286
8.273.2.2 pDualStandByPrefInd . . . 287
8.273.2.3 pErrorRateInd . . . 287
8.273.2.4 pHDRNewUATIAssInd . . . 287
8.273.2.5 pHDRSessionCloseInd . . . 287
8.273.2.6 pLTECphyCa . . . 287
8.273.2.7 pManagedRoamingInd . . . 287
8.273.2.8 pNetworkRejectInd . . . 287
8.273.2.9 pNetworkTimeInd . . . 287
8.273.2.10 pServingSystemInd . . . 287
8.273.2.11 pSignalStrengthInd . . . 287
8.273.2.12 pSubscriptionInfoInd . . . 287
8.273.2.13 pSuppressSysInfoInd . . . 287
8.273.2.14 pSysInfoInd . . . 287
8.273.2.15 pSystemSelectionInd . . . 287
8.274 pack_nas_SLQSNasSwiIndicationRegister_t Struct Reference . . . 287
8.274.1 Detailed Description . . . 287
8.274.2 Field Documentation . . . 288
8.274.2.1 gsmUmtsDl . . . 288

8.274.2.2 gsmUmtsUl . . . 288
8.274.2.3 lteEmmDl . . . 288
8.274.2.4 lteEmmUl . . . 288
8.274.2.5 lteEsmDl . . . 288
8.274.2.6 lteEsmUl . . . 288
8.274.2.7 pRankIndicatorInd . . . 288
8.274.2.8 pTimer . . . 288
8.275 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference . . . 288
8.275.1 Detailed Description . . . 289
8.275.2 Field Documentation . . . 289
8.275.2.1 pLteEARFCN . . . 289
8.275.2.2 pLtePCI . . . 289
8.275.2.3 pWcdmaUARFCN . . . 289
8.276 pack_nas_SLQSSetBandPreference_t Struct Reference . . . 289
8.276.1 Detailed Description . . . 289
8.276.2 Field Documentation . . . 290
8.276.2.1 bandPref . . . 291
8.277 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference . . . 291
8.277.1 Detailed Description . . . 291
8.277.2 Field Documentation . . . 291
8.277.2.1 bEnable . . . 291
8.277.2.2 pSigIndReq . . . 291
8.278 pack_nas_SLQSSetSysSelectionPref_t Struct Reference . . . 291
8.278.1 Detailed Description . . . 291
8.278.2 Field Documentation . . . 296
8.278.2.1 pAcqOrderPref . . . 296
8.278.2.2 pBandPref . . . 296
8.278.2.3 pChgDuration . . . 296
8.278.2.4 pCSGID . . . 296
8.278.2.5 pEmerMode . . . 296
8.278.2.6 pGWAcqOrderPref . . . 296
8.278.2.7 pLTEBandPref . . . 296
8.278.2.8 pMNCIncPCSDigStat . . . 296
8.278.2.9 pModePref . . . 296
8.278.2.10 pNetSelPref . . . 296
8.278.2.11 pPRLPref . . . 296
8.278.2.12 pRAT . . . 296
8.278.2.13 pRoamPref . . . 296
8.278.2.14 pSrvDomainPref . . . 296
8.278.2.15 pSrvRegRestriction . . . 297

8.278.2.16 pTdscdmaBandPref . . . 297
8.279 pack_nas_SLQSSwiPSDetach_t Struct Reference . . . 297
8.279.1 Detailed Description . . . 297
8.279.2 Field Documentation . . . 297
8.279.2.1 pDetachAction . . . 297
8.280 pack_pds_PDSInjectTimeReference_t Struct Reference . . . 297
8.280.1 Detailed Description . . . 297
8.280.2 Field Documentation . . . 297
8.280.2.1 systemDiscontinuities . . . 297
8.280.2.2 systemTime . . . 297
8.281 pack_pds_ResetPDSData_t Struct Reference . . . 298
8.281.1 Detailed Description . . . 298
8.281.2 Field Documentation . . . 298
8.281.2.1 pCellDataMask . . . 298
8.281.2.2 pGPSDataMask . . . 298
8.282 pack_pds_SetEventReportCallback_t Struct Reference . . . 298
8.282.1 Detailed Description . . . 299
8.282.2 Field Documentation . . . 299
8.282.2.1 posDataNmea . . . 299
8.282.2.2 rptPosData . . . 299
8.283 pack_pds_SetPDSDefaults_t Struct Reference . . . 299
8.283.1 Detailed Description . . . 299
8.283.2 Field Documentation . . . 300
8.283.2.1 accuracy . . . 300
8.283.2.2 interval . . . 300
8.283.2.3 operation . . . 300
8.283.2.4 timeout . . . 300
8.284 pack_pds_SetPDSState_t Struct Reference . . . 300
8.284.1 Detailed Description . . . 300
8.284.2 Field Documentation . . . 300
8.284.2.1 enable . . . 300
8.285 pack_pds_SetPortAutomaticTracking_t Struct Reference . . . 300
8.285.1 Detailed Description . . . 300
8.285.2 Field Documentation . . . 301
8.285.2.1 bAuto . . . 301
8.286 pack_pds_SetServiceAutomaticTracking_t Struct Reference . . . 301
8.286.1 Detailed Description . . . 301
8.286.2 Field Documentation . . . 301
8.286.2.1 bAuto . . . 301
8.287 pack_pds_SetXTRAAutomaticDownload_t Struct Reference . . . 301

8.287.1 Detailed Description . . . 301
8.287.2 Field Documentation . . . 302
8.287.2.1 bEnabled . . . 302
8.287.2.2 interval . . . 302
8.288 pack_pds_SetXTRANetwork_t Struct Reference . . . 302
8.288.1 Detailed Description . . . 302
8.288.2 Field Documentation . . . 302
8.288.2.1 preference . . . 302
8.289 pack_pds_SLQSGetAGPSConfig_t Struct Reference . . . 302
8.289.1 Detailed Description . . . 302
8.289.2 Field Documentation . . . 303
8.289.2.1 pNetworkMode . . . 303
8.290 pack_pds_SLQSPDSInjectAbsoluteTimeReference_t Struct Reference . . . 303
8.290.1 Detailed Description . . . 303
8.290.2 Field Documentation . . . 303
8.290.2.1 forceFlag . . . 303
8.290.2.2 timeBase . . . 303
8.290.2.3 timeMsec . . . 303
8.290.2.4 timeUncMsec . . . 303
8.291 pack_pds_SLQSPDSInjectPositionData_t Struct Reference . . . 303
8.291.1 Detailed Description . . . 304
8.291.2 Field Documentation . . . 305
8.291.2.1 pAltitudeWrtEllipsoid . . . 305
8.291.2.2 pAltitudeWrtSealevel . . . 305
8.291.2.3 pHorizontalConfidence . . . 305
8.291.2.4 pHorizontalUncCircular . . . 305
8.291.2.5 pLatitude . . . 305
8.291.2.6 pLongitude . . . 305
8.291.2.7 pPositionSource . . . 305
8.291.2.8 pTimeStamp . . . 305
8.291.2.9 pTimeType . . . 305
8.291.2.10 pVerticalConfidence . . . 305
8.291.2.11 pVerticalUnc . . . 305
8.292 pack_pds_SLQSSetAGPSConfig_t Struct Reference . . . 305
8.292.1 Detailed Description . . . 306
8.292.2 Field Documentation . . . 306
8.292.2.1 pNetworkMode . . . 306
8.292.2.2 pServerAddress . . . 306
8.292.2.3 pServerPort . . . 306
8.292.2.4 pServerURL . . . 306

8.292.2.5 pServerURLLength . . . 306
8.293 pack_pds_SLQSSetPositionMethodState_t Struct Reference . . . 306
8.293.1 Detailed Description . . . 306
8.293.2 Field Documentation . . . 307
8.293.2.1 pWifiState . . . 307
8.293.2.2 pXtraDataState . . . 307
8.293.2.3 pXtraTimeState . . . 307
8.294 pack_pds_StartPDSTrackingSessionExt_t Struct Reference . . . 307
8.294.1 Detailed Description . . . 307
8.294.2 Field Documentation . . . 308
8.294.2.1 fixAccuracy . . . 308
8.294.2.2 fixCount . . . 308
8.294.2.3 fixInterval . . . 308
8.294.2.4 fixTimeout . . . 308
8.294.2.5 sessionControl . . . 308
8.294.2.6 sessionOperation . . . 308
8.294.2.7 sessionServerOption . . . 308
8.294.2.8 sessionType . . . 308
8.295 pack_qmi_t Struct Reference . . . 308
8.295.1 Detailed Description . . . 309
8.295.2 Field Documentation . . . 309
8.295.2.1 msgid . . . 309
8.295.2.2 svc . . . 309
8.295.2.3 timeout . . . 309
8.295.2.4 xid . . . 309
8.296 pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 309
8.296.1 Detailed Description . . . 309
8.296.2 Field Documentation . . . 309
8.296.2.1 apnId . . . 309
8.297 pack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 309
8.297.1 Detailed Description . . . 309
8.297.2 Field Documentation . . . 310
8.297.2.1 apnId . . . 310
8.298 pack_qos_SLQSSetQosEventCallback_t Struct Reference . . . 310
8.298.1 Detailed Description . . . 310
8.298.2 Field Documentation . . . 310
8.298.2.1 enable . . . 310
8.299 pack_rms_SetSMSWake_t Struct Reference . . . 310
8.299.1 Detailed Description . . . 310
8.299.2 Field Documentation . . . 310

8.299.2.1 enabled . . . 311
8.299.2.2 wake_mask . . . 311
8.300 pack_sar_SLQSSetRfSarState_t Struct Reference . . . 311
8.300.1 Detailed Description . . . 311
8.300.2 Field Documentation . . . 311
8.300.2.1 RfSarState . . . 311
8.301 pack_sms_SaveSMS_t Struct Reference . . . 311
8.301.1 Detailed Description . . . 311
8.301.2 Field Documentation . . . 312
8.301.2.1 messageFormat . . . 312
8.301.2.2 messageSize . . . 312
8.301.2.3 pMessage . . . 312
8.301.2.4 storageType . . . 312
8.302 pack_sms_SendSMS_t Struct Reference . . . 312
8.302.1 Detailed Description . . . 312
8.302.2 Field Documentation . . . 313
8.302.2.1 messageFormat . . . 313
8.302.2.2 messageSize . . . 313
8.302.2.3 pLinktimer . . . 313
8.302.2.4 pMessage . . . 313
8.303 pack_sms_SetNewSMSCallback_t Struct Reference . . . 313
8.303.1 Detailed Description . . . 313
8.303.2 Field Documentation . . . 313
8.303.2.1 status . . . 313
8.304 pack_sms_SetSMSCAddress_t Struct Reference . . . 313
8.304.1 Detailed Description . . . 313
8.304.2 Field Documentation . . . 313
8.304.2.1 pSMSCAddress . . . 313
8.304.2.2 pSMSCType . . . 313
8.305 pack_sms_SLQSDeleteSMS_t Struct Reference . . . 314
8.305.1 Detailed Description . . . 314
8.305.2 Field Documentation . . . 314
8.305.2.1 pMessageIndex . . . 314
8.305.2.2 pMessageMode . . . 314
8.305.2.3 pMessageTag . . . 314
8.305.2.4 storageType . . . 314
8.306 pack_sms_SLQSGetSMS_t Struct Reference . . . 314
8.306.1 Detailed Description . . . 315
8.306.2 Field Documentation . . . 315
8.306.2.1 messageIndex . . . 315

8.306.2.2 pMessageMode . . . 315
8.306.2.3 storageType . . . 315
8.307 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 315
8.307.1 Detailed Description . . . 315
8.307.2 Field Documentation . . . 315
8.307.2.1 mode . . . 315
8.308 pack_sms_SLQSGetSMSList_t Struct Reference . . . 315
8.308.1 Detailed Description . . . 316
8.308.2 Field Documentation . . . 316
8.308.2.1 pMessageMode . . . 316
8.308.2.2 pRequestedTag . . . 316
8.308.2.3 storageType . . . 316
8.309 pack_sms_SLQSModifySMSStatus_t Struct Reference . . . 316
8.309.1 Detailed Description . . . 316
8.309.2 Field Documentation . . . 317
8.309.2.1 messageIndex . . . 317
8.309.2.2 messageTag . . . 317
8.309.2.3 pMessageMode . . . 317
8.309.2.4 storageType . . . 317
8.310 pack_sms_SLQSSendAsyncSMS_t Struct Reference . . . 317
8.310.1 Detailed Description . . . 317
8.310.2 Field Documentation . . . 317
8.310.2.1 pSendSmsParams . . . 317
8.311 pack_sms_SLQSSetIndicationRegister_t Struct Reference . . . 317
8.311.1 Detailed Description . . . 317
8.311.2 Field Documentation . . . 318
8.311.2.1 pSetIndicationRegReq . . . 318
8.312 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference . . . 318
8.312.1 Detailed Description . . . 318
8.312.2 Field Documentation . . . 318
8.312.2.1 broadcastActivate . . . 318
8.312.2.2 mode . . . 318
8.313 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference . . . 318
8.313.1 Detailed Description . . . 318
8.313.2 Field Documentation . . . 319
8.313.2.1 mode . . . 319
8.313.2.2 pBroadcastConfig . . . 319
8.313.2.3 pCDMABroadcastConfig . . . 319
8.314 pack_sms_SLQSSetSmsStorage_t Struct Reference . . . 319
8.314.1 Detailed Description . . . 319

8.314.2 Field Documentation . . . 319
8.314.2.1 smsStorage . . . 319
8.315 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 319
8.315.1 Detailed Description . . . 319
8.315.2 Field Documentation . . . 320
8.315.2.1 pMaxStorageSizeReq . . . 320
8.316 pack_sms_SLQSSmsSetRoutes_t Struct Reference . . . 320
8.316.1 Detailed Description . . . 320
8.316.2 Field Documentation . . . 320
8.316.2.1 pSetRoutesReq . . . 320
8.317 pack_swiavms_SLQSAVMSSendSelection_t Struct Reference . . . 320
8.317.1 Detailed Description . . . 320
8.317.2 Field Documentation . . . 321
8.317.2.1 pClientPerformOperationFlag . . . 321
8.317.2.2 pDeferTime . . . 321
8.317.2.3 pPackageID . . . 321
8.317.2.4 pRejectReason . . . 321
8.317.2.5 selection . . . 321
8.318 pack_swiavms_SLQSAVMSSetSettings_t Struct Reference . . . 321
8.318.1 Detailed Description . . . 321
8.318.2 Field Documentation . . . 322
8.318.2.1 AutoConnect . . . 322
8.318.2.2 AutoReboot . . . 322
8.318.2.3 FwAutodownload . . . 322
8.318.2.4 FwAutoUpdate . . . 323
8.318.2.5 pAPNInfo . . . 323
8.318.2.6 pConnectionRetryTimers . . . 323
8.318.2.7 pFwAutoSDM . . . 323
8.318.2.8 pNotifStore . . . 323
8.318.2.9 pPeriodInfo . . . 323
8.318.2.10 pPollingTimer . . . 323
8.319 pack_swiavms_SLQSAVMSStartSession_t Struct Reference . . . 323
8.319.1 Detailed Description . . . 323
8.319.2 Field Documentation . . . 323
8.319.2.1 sessionType . . . 323
8.320 pack_swiavms_SLQSAVMSStopSession_t Struct Reference . . . 323
8.320.1 Detailed Description . . . 323
8.320.2 Field Documentation . . . 324
8.320.2.1 sessionType . . . 324
8.321 pack_swiloc_SwiLocSetAutoStart_t Struct Reference . . . 324

8.321.1 Detailed Description . . . 324
8.321.2 Field Documentation . . . 325
8.321.2.1 fix_rate . . . 325
8.321.2.2 fix_type . . . 325
8.321.2.3 function . . . 325
8.321.2.4 max_dist . . . 325
8.321.2.5 max_time . . . 325
8.321.2.6 set_fix_rate . . . 325
8.321.2.7 set_fix_type . . . 325
8.321.2.8 set_function . . . 325
8.321.2.9 set_max_dist . . . 325
8.321.2.10 set_max_time . . . 325
8.322 pack_swioma_SLQSOMADMCancelSession_t Struct Reference . . . 325
8.322.1 Detailed Description . . . 326
8.322.2 Field Documentation . . . 326
8.322.2.1 sessionType . . . 326
8.323 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 326
8.323.1 Detailed Description . . . 326
8.323.2 Field Documentation . . . 326
8.323.2.1 SessionType . . . 326
8.324 pack_swioma_SLQSOMADMSendSelection_t Struct Reference . . . 326
8.324.1 Detailed Description . . . 327
8.324.2 Field Documentation . . . 327
8.324.2.1 pDeferTime . . . 327
8.324.2.2 pRejectReason . . . 327
8.324.2.3 selection . . . 327
8.325 pack_swioma_SLQSOMADMSetSettings_t Struct Reference . . . 327
8.325.1 Detailed Description . . . 327
8.325.2 Field Documentation . . . 328
8.325.2.1 FOTAdownload . . . 328
8.325.2.2 FOTAUpdate . . . 328
8.325.2.3 pAutosdm . . . 328
8.325.2.4 pFwAutoCheck . . . 328
8.326 pack_swioma_SLQSOMADMStartSession_t Struct Reference . . . 328
8.326.1 Detailed Description . . . 328
8.326.2 Field Documentation . . . 328
8.326.2.1 sessionType . . . 328
8.327 pack_uim_ChangePin_t Struct Reference . . . 329
8.327.1 Detailed Description . . . 329
8.327.2 Field Documentation . . . 329

8.327.2.1 changePIN . . . 329
8.327.2.2 EncryptedPIN1 . . . 329
8.327.2.3 pIndicationToken . . . 329
8.327.2.4 pKeyReferenceID . . . 329
8.327.2.5 sessionInfo . . . 329
8.327.2.6 Tlvresult . . . 330
8.328 pack_uim_ReadTransparent_t Struct Reference . . . 330
8.328.1 Detailed Description . . . 330
8.328.2 Field Documentation . . . 330
8.328.2.1 fileIndex . . . 330
8.328.2.2 pEncryptData . . . 330
8.328.2.3 pIndicationToken . . . 330
8.328.2.4 readTransparent . . . 331
8.328.2.5 sessionInfo . . . 331
8.328.2.6 Tlvresult . . . 331
8.329 pack_uim_SetPinProtection_t Struct Reference . . . 331
8.329.1 Detailed Description . . . 331
8.329.2 Field Documentation . . . 331
8.329.2.1 EncryptedPIN1 . . . 331
8.329.2.2 pIndicationToken . . . 331
8.329.2.3 pinProtection . . . 332
8.329.2.4 pKeyReferenceID . . . 332
8.329.2.5 sessionInfo . . . 332
8.329.2.6 Tlvresult . . . 332
8.330 pack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 332
8.330.1 Detailed Description . . . 332
8.330.2 Field Documentation . . . 332
8.330.2.1 authData . . . 332
8.330.2.2 pIndicationToken . . . 332
8.330.2.3 sessionInfo . . . 332
8.331 pack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 332
8.331.1 Detailed Description . . . 333
8.331.2 Field Documentation . . . 333
8.331.2.1 depersonilisationInfo . . . 333
8.332 pack_uim_SLQSUIMEventRegister_t Struct Reference . . . 333
8.332.1 Detailed Description . . . 333
8.332.2 Field Documentation . . . 333
8.332.2.1 eventMask . . . 333
8.333 pack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 333
8.333.1 Detailed Description . . . 333

8.333.2 Field Documentation . . . 334
8.333.2.1 pConfigurationMask . . . 334
8.334 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 334
8.334.1 Detailed Description . . . 334
8.334.2 Field Documentation . . . 334
8.334.2.1 fileIndex . . . 334
8.334.2.2 pIndicationToken . . . 334
8.334.2.3 sessionInfo . . . 334
8.335 pack_uim_SLQSUIMPowerDown_t Struct Reference . . . 334
8.335.1 Detailed Description . . . 335
8.335.2 Field Documentation . . . 335
8.335.2.1 slot . . . 335
8.336 pack_uim_SLQSUIMPowerUp_t Struct Reference . . . 335
8.336.1 Detailed Description . . . 335
8.336.2 Field Documentation . . . 335
8.336.2.1 pIgnoreHotSwapSwitch . . . 335
8.336.2.2 slot . . . 335
8.337 pack_uim_SLQSUIMRefreshComplete_t Struct Reference . . . 335
8.337.1 Detailed Description . . . 336
8.337.2 Field Documentation . . . 336
8.337.2.1 refreshComplete . . . 336
8.337.2.2 sessionInfo . . . 336
8.338 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 336
8.338.1 Detailed Description . . . 336
8.338.2 Field Documentation . . . 336
8.338.2.1 sessionInfo . . . 336
8.339 pack_uim_SLQSUIMRefreshOK_t Struct Reference . . . 336
8.339.1 Detailed Description . . . 336
8.339.2 Field Documentation . . . 337
8.339.2.1 OKtoRefresh . . . 337
8.339.2.2 sessionInfo . . . 337
8.340 pack_uim_SLQSUIMRefreshRegister_t Struct Reference . . . 337
8.340.1 Detailed Description . . . 337
8.340.2 Field Documentation . . . 337
8.340.2.1 regRefresh . . . 337
8.340.2.2 sessionInfo . . . 337
8.341 pack_uim_SLQSUIMSwitchSlot_t Struct Reference . . . 337
8.341.1 Detailed Description . . . 338
8.341.2 Field Documentation . . . 338
8.341.2.1 bLogicalSlot . . . 338

8.341.2.2 ulPhysicalSlot . . . 338
8.342 pack_uim_UnblockPin_t Struct Reference . . . 338
8.342.1 Detailed Description . . . 338
8.342.2 Field Documentation . . . 339
8.342.2.1 EncryptedPIN1 . . . 339
8.342.2.2 pIndicationToken . . . 339
8.342.2.3 pinProtection . . . 339
8.342.2.4 pKeyReferenceID . . . 339
8.342.2.5 sessionInfo . . . 339
8.342.2.6 Tlvresult . . . 339
8.343 pack_uim_VerifyPin_t Struct Reference . . . 339
8.343.1 Detailed Description . . . 339
8.343.2 Field Documentation . . . 340
8.343.2.1 pEncryptedPIN1 . . . 340
8.343.2.2 pIndicationToken . . . 340
8.343.2.3 pKeyReferenceID . . . 340
8.343.2.4 sessionInfo . . . 340
8.343.2.5 Tlvresult . . . 340
8.343.2.6 verifyPIN . . . 340
8.344 pack_voice_AnswerUSSD_t Struct Reference . . . 340
8.344.1 Detailed Description . . . 340
8.344.2 Field Documentation . . . 341
8.344.2.1 pInfo . . . 341
8.345 pack_voice_OriginateUSSD_t Struct Reference . . . 341
8.345.1 Detailed Description . . . 341
8.345.2 Field Documentation . . . 341
8.345.2.1 pInfo . . . 341
8.346 pack_voice_SLQSOriginateUSSD_t Struct Reference . . . 341
8.346.1 Detailed Description . . . 341
8.346.2 Field Documentation . . . 341
8.346.2.1 ussData . . . 342
8.346.2.2 ussDCS . . . 342
8.346.2.3 ussLen . . . 342
8.347 pack_voice_SLQSVoiceALSSelectLine_t Struct Reference . . . 342
8.347.1 Detailed Description . . . 342
8.347.2 Field Documentation . . . 342
8.347.2.1 lineValue . . . 342
8.348 pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference . . . 342
8.348.1 Detailed Description . . . 342
8.348.2 Field Documentation . . . 342

8.348.2.1 switchOption . . . 342
8.349 pack_voice_SLQSVoiceAnswerCall_t Struct Reference . . . 342
8.349.1 Detailed Description . . . 343
8.349.2 Field Documentation . . . 343
8.349.2.1 pCallId . . . 343
8.350 pack_voice_SLQSVoiceBindSubscription_t Struct Reference . . . 343
8.350.1 Detailed Description . . . 343
8.350.2 Field Documentation . . . 343
8.350.2.1 subsType . . . 343
8.351 pack_voice_SLQSVoiceBurstDTMF_t Struct Reference . . . 343
8.351.1 Detailed Description . . . 343
8.351.2 Field Documentation . . . 344
8.351.2.1 BurstDTMFInfo . . . 344
8.351.2.2 pBurstDTMFLengths . . . 344
8.352 pack_voice_SLQSVoiceDialCall_t Struct Reference . . . 344
8.352.1 Detailed Description . . . 344
8.352.2 Field Documentation . . . 345
8.352.2.1 callNumber . . . 345
8.352.2.2 pCallPartySubAdd . . . 345
8.352.2.3 pCallType . . . 345
8.352.2.4 pCLIRType . . . 345
8.352.2.5 pCUGInfo . . . 345
8.352.2.6 pEmergencyCategory . . . 345
8.352.2.7 pSvcType . . . 345
8.352.2.8 pUUSInFo . . . 345
8.353 pack_voice_SLQSVoiceEndCall_t Struct Reference . . . 346
8.353.1 Detailed Description . . . 346
8.353.2 Field Documentation . . . 346
8.353.2.1 pCallId . . . 346
8.354 pack_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 346
8.354.1 Detailed Description . . . 346
8.354.2 Field Documentation . . . 347
8.354.2.1 pSvcClass . . . 347
8.354.2.2 reason . . . 347
8.355 pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 347
8.355.1 Detailed Description . . . 347
8.355.2 Field Documentation . . . 347
8.355.2.1 pSvcClass . . . 347
8.355.2.2 Reason . . . 347
8.356 pack_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 347

8.356.1 Detailed Description . . . 348
8.356.2 Field Documentation . . . 348
8.356.2.1 callID . . . 348
8.357pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 348
8.357.1 Detailed Description . . . 348
8.357.2 Field Documentation . . . 348
8.357.2.1 pSvcClass . . . 348
8.358pack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 348
8.358.1 Detailed Description . . . 349
8.358.2 Field Documentation . . . 350
8.358.2.1 pAirTimer . . . 350
8.358.2.2 pAMRStatus . . . 350
8.358.2.3 pAutoAnswer . . . 350
8.358.2.4 pNamID . . . 350
8.358.2.5 pPrefVoicePrivacy . . . 350
8.358.2.6 pPrefVoiceSO . . . 350
8.358.2.7 pRoamTimer . . . 350
8.358.2.8 pTTYMode . . . 350
8.358.2.9 pVoiceDomainPref . . . 350
8.359pack_voice_SLQSVoiceIndicationRegister_t Struct Reference . . . 350
8.359.1 Detailed Description . . . 350
8.359.2 Field Documentation . . . 351
8.359.2.1 pRegDTMFEvents . . . 351
8.359.2.2 pRegVoicePrivacyEvents . . . 351
8.359.2.3 pSuppsNotifEvents . . . 351
8.360pack_voice_SLQSVoiceManageCalls_t Struct Reference . . . 351
8.360.1 Detailed Description . . . 351
8.360.2 Field Documentation . . . 351
8.360.2.1 pCallID . . . 352
8.360.2.2 SUPSType . . . 352
8.361pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference . . . 352
8.361.1 Detailed Description . . . 352
8.361.2 Field Documentation . . . 352
8.361.2.1 USSInformation . . . 352
8.362pack_voice_SLQSVoiceSendFlash_t Struct Reference . . . 352
8.362.1 Detailed Description . . . 352
8.362.2 Field Documentation . . . 353
8.362.2.1 pCallID . . . 353
8.362.2.2 pFlashPayLd . . . 353
8.362.2.3 pFlashType . . . 353

8.363pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 353
8.363.1 Detailed Description . . . 353
8.363.2 Field Documentation . . . 354
8.363.2.1 newPasswd . . . 354
8.363.2.2 newPasswdAgain . . . 354
8.363.2.3 oldPasswd . . . 354
8.363.2.4 Reason . . . 354
8.364pack_voice_SLQSVoiceSetConfig_t Struct Reference . . . 354
8.364.1 Detailed Description . . . 354
8.364.2 Field Documentation . . . 355
8.364.2.1 pAirTimerConfig . . . 355
8.364.2.2 pAutoAnswer . . . 355
8.364.2.3 pPrefVoiceDomain . . . 355
8.364.2.4 pPrefVoiceSO . . . 355
8.364.2.5 pRoamTimerConfig . . . 355
8.364.2.6 pTTYMode . . . 355
8.365pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference . . . 355
8.365.1 Detailed Description . . . 355
8.365.2 Field Documentation . . . 355
8.365.2.1 privacyPref . . . 355
8.366pack_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 355
8.366.1 Detailed Description . . . 356
8.366.2 Field Documentation . . . 358
8.366.2.1 pCallBarringPasswd . . . 358
8.366.2.2 pCallForwardingNumber . . . 358
8.366.2.3 pCallFwdTypeAndPlan . . . 358
8.366.2.4 pServiceClass . . . 358
8.366.2.5 pTimerVal . . . 358
8.366.2.6 reason . . . 358
8.366.2.7 voiceSvc . . . 358
8.367pack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 358
8.367.1 Detailed Description . . . 358
8.367.2 Field Documentation . . . 358
8.367.2.1 DTMFdigit . . . 358
8.367.2.2 pCallID . . . 359
8.368pack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 359
8.368.1 Detailed Description . . . 359
8.368.2 Field Documentation . . . 359
8.368.2.1 callID . . . 359
8.369pack_wds_DHCPv4ClientLeaseChange_t Struct Reference . . . 359

8.369.1 Detailed Description . . . 359
8.369.2 Field Documentation . . . 359
8.369.2.1 pEnableNotification . . . 359
8.370 pack_wds_GetDefaultProfile_t Struct Reference . . . 359
8.370.1 Detailed Description . . . 360
8.370.2 Field Documentation . . . 360
8.370.2.1 profiletype . . . 360
8.371 pack_wds_GetDefaultProfileNum_t Struct Reference . . . 360
8.371.1 Detailed Description . . . 360
8.371.2 Field Documentation . . . 360
8.371.2.1 family . . . 360
8.371.2.2 type . . . 360
8.372 pack_wds_GetDormancyState_t Struct Reference . . . 360
8.372.1 Detailed Description . . . 360
8.373 pack_wds_GetLastMobileIPError_t Struct Reference . . . 360
8.373.1 Detailed Description . . . 360
8.374 pack_wds_GetMobileIP_t Struct Reference . . . 361
8.374.1 Detailed Description . . . 361
8.375 pack_wds_GetMobileIPProfile_t Struct Reference . . . 361
8.375.1 Detailed Description . . . 361
8.375.2 Field Documentation . . . 361
8.375.2.1 index . . . 361
8.376 pack_wds_GetPacketStatistics_t Struct Reference . . . 361
8.376.1 Detailed Description . . . 361
8.376.2 Field Documentation . . . 362
8.376.2.1 pStatMask . . . 362
8.377 pack_wds_GetPacketStatus_t Struct Reference . . . 362
8.377.1 Detailed Description . . . 362
8.377.2 Field Documentation . . . 362
8.377.2.1 statmask . . . 362
8.378 pack_wds_GetSessionDuration_t Struct Reference . . . 362
8.378.1 Detailed Description . . . 362
8.379 pack_wds_RMSetTransferStatistics_t Struct Reference . . . 362
8.379.1 Detailed Description . . . 363
8.379.2 Field Documentation . . . 363
8.379.2.1 RmTrasnferStaticsReq . . . 363
8.380 pack_wds_SetAutoconnect_t Struct Reference . . . 363
8.380.1 Detailed Description . . . 363
8.380.2 Field Documentation . . . 363
8.380.2.1 acroamsetting . . . 363

8.380.2.2 acsetting . . . 363
8.381 pack_wds_SetDefaultProfile_t Struct Reference . . . 363
8.381.1 Detailed Description . . . 364
8.381.2 Field Documentation . . . 365
8.381.2.1 authentication . . . 365
8.381.2.2 ipAddress . . . 365
8.381.2.3 pApnname . . . 365
8.381.2.4 pdpType . . . 365
8.381.2.5 pName . . . 365
8.381.2.6 pPassword . . . 365
8.381.2.7 primaryDNS . . . 365
8.381.2.8 profileType . . . 365
8.381.2.9 pUsername . . . 365
8.381.2.10 secondaryDNS . . . 365
8.382 pack_wds_SetDefaultProfileNum_t Struct Reference . . . 365
8.382.1 Detailed Description . . . 365
8.382.2 Field Documentation . . . 366
8.382.2.1 family . . . 366
8.382.2.2 index . . . 366
8.382.2.3 type . . . 366
8.383 pack_wds_SetMobileIP_t Struct Reference . . . 366
8.383.1 Detailed Description . . . 366
8.383.2 Field Documentation . . . 366
8.383.2.1 mode . . . 366
8.384 pack_wds_SetMobileIPParameters_t Struct Reference . . . 366
8.384.1 Detailed Description . . . 366
8.384.2 Field Documentation . . . 367
8.384.2.1 pHA2002bis . . . 367
8.384.2.2 pHAAuthenticator . . . 367
8.384.2.3 pMode . . . 367
8.384.2.4 pReRegPeriod . . . 367
8.384.2.5 pReRegTraffic . . . 367
8.384.2.6 pRetryInterval . . . 367
8.384.2.7 pRetryLimit . . . 367
8.384.2.8 pSPC . . . 367
8.385 pack_wds_SetMobileIPProfile_t Struct Reference . . . 367
8.385.1 Detailed Description . . . 368
8.385.2 Field Documentation . . . 369
8.385.2.1 index . . . 369
8.385.2.2 pAAASPI . . . 369

8.385.2.3 pAddress . . . 369
8.385.2.4 pEnabled . . . 369
8.385.2.5 pHASPI . . . 369
8.385.2.6 pMNAAA . . . 369
8.385.2.7 pMNHA . . . 369
8.385.2.8 pNAI . . . 369
8.385.2.9 pPrimaryHA . . . 369
8.385.2.10 pRevTunneling . . . 369
8.385.2.11 pSecondaryHA . . . 369
8.385.2.12 spc . . . 369
8.386 pack_wds_SLQSCreateProfile_t Struct Reference . . . 369
8.386.1 Detailed Description . . . 369
8.386.2 Field Documentation . . . 370
8.386.2.1 pCurProfile . . . 370
8.386.2.2 pProfileId . . . 370
8.386.2.3 pProfileType . . . 370
8.387 pack_wds_SLQSDeleteProfile_t Struct Reference . . . 370
8.387.1 Detailed Description . . . 370
8.387.2 Field Documentation . . . 370
8.387.2.1 profileIndex . . . 370
8.387.2.2 profileType . . . 370
8.388 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 370
8.388.1 Detailed Description . . . 370
8.389 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 371
8.389.1 Detailed Description . . . 371
8.390 pack_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 371
8.390.1 Detailed Description . . . 371
8.390.2 Field Documentation . . . 372
8.390.2.1 Mask . . . 372
8.390.2.2 pReportChannelRate . . . 372
8.390.2.3 pReportConnStatus . . . 372
8.390.2.4 pReportDataBearerTech . . . 372
8.390.2.5 pReportDormStatus . . . 372
8.390.2.6 pTransferStatInd . . . 372
8.391 pack_wds_SLQSGetProfileSettings_t Struct Reference . . . 372
8.391.1 Detailed Description . . . 372
8.391.2 Field Documentation . . . 373
8.391.2.1 ProfileId . . . 373
8.391.2.2 ProfileType . . . 373
8.392 pack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 373

8.392.1 Detailed Description . . . 373
8.392.2 Field Documentation . . . 374
8.392.2.1 pReqSettings . . . 374
8.393pack_wds_SLQSModifyProfile_t Struct Reference . . . 374
8.393.1 Detailed Description . . . 374
8.393.2 Field Documentation . . . 375
8.393.2.1 curProfile . . . 375
8.393.2.2 pProfileId . . . 375
8.393.2.3 pProfileType . . . 375
8.394pack_wds_SLQSSet3GPPConfigItem_t Struct Reference . . . 375
8.394.1 Detailed Description . . . 375
8.394.2 Field Documentation . . . 376
8.394.2.1 LTEAttachProfileListLen . . . 376
8.394.2.2 p3gppRelease . . . 376
8.394.2.3 pDefaultPDNEnabled . . . 376
8.394.2.4 pLTEAttachProfile . . . 376
8.394.2.5 pLTEAttachProfileList . . . 376
8.394.2.6 pProfileList . . . 377
8.395pack_wds_SLQSSetIPFamilyPreference_t Struct Reference . . . 377
8.395.1 Detailed Description . . . 377
8.395.2 Field Documentation . . . 377
8.395.2.1 IPFamilyPreference . . . 377
8.396pack_wds_SLQSSetWdsEventCallback_t Struct Reference . . . 377
8.396.1 Detailed Description . . . 377
8.396.2 Field Documentation . . . 377
8.396.2.1 currentDataBearer . . . 378
8.396.2.2 dataBearer . . . 378
8.396.2.3 dataBearerTechExt . . . 378
8.396.2.4 dataSystemStatus . . . 378
8.396.2.5 dormancyStatus . . . 378
8.396.2.6 interval . . . 378
8.396.2.7 mobileIP . . . 378
8.396.2.8 transferStats . . . 378
8.397pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 378
8.397.1 Detailed Description . . . 378
8.397.2 Field Documentation . . . 378
8.397.2.1 pProfileId . . . 378
8.398pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference . . . 378
8.398.1 Detailed Description . . . 378
8.398.2 Field Documentation . . . 379

8.398.2.1 pHwConfig . . . 379
8.398.2.2 pProfileId . . . 379
8.398.2.3 pRequestOptionList . . . 379
8.399 pack_wds_SLQSSSetLoopback_t Struct Reference . . . 379
8.399.1 Detailed Description . . . 379
8.399.2 Field Documentation . . . 379
8.399.2.1 loopbackMode . . . 379
8.399.2.2 loopbackMultiplier . . . 379
8.400 pack_wds_SLQSStartDataSession_t Struct Reference . . . 379
8.400.1 Detailed Description . . . 379
8.400.2 Field Documentation . . . 380
8.400.2.1 pAuth . . . 380
8.400.2.2 pPass . . . 380
8.400.2.3 pprofileid3gpp . . . 380
8.400.2.4 pprofileid3gpp2 . . . 380
8.400.2.5 pTech . . . 380
8.400.2.6 pUser . . . 380
8.401 pack_wds_SLQSStopDataSession_t Struct Reference . . . 380
8.401.1 Detailed Description . . . 381
8.401.2 Field Documentation . . . 381
8.401.2.1 psid . . . 381
8.402 pack_wds_SLQSWdsSetEventReport_t Struct Reference . . . 381
8.402.1 Detailed Description . . . 381
8.402.2 Field Documentation . . . 382
8.402.2.1 pCurrChannelRateInd . . . 382
8.402.2.2 pCurrDataBearerTechInd . . . 382
8.402.2.3 pCurrPrefDataSysInd . . . 382
8.402.2.4 pDataBearerTechInd . . . 382
8.402.2.5 pDataCallStatusChangeInd . . . 382
8.402.2.6 pDataSystemStatusChangeInd . . . 382
8.402.2.7 pDormancyStatusInd . . . 382
8.402.2.8 pEVDOPageMonPerChangeInd . . . 382
8.402.2.9 pMIPStatusInd . . . 382
8.402.2.10 pTransferStatInd . . . 382
8.403 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 383
8.403.1 Detailed Description . . . 383
8.403.2 Field Documentation . . . 383
8.403.2.1 contextId . . . 383
8.403.2.2 contextType . . . 383
8.404 PackCreateProfileOut Struct Reference . . . 383

8.404.1 Detailed Description . . . 383
8.404.2 Field Documentation . . . 384
8.404.2.1 ExtErrorCode . . . 384
8.404.2.2 ProfileIndex . . . 384
8.404.2.3 ProfileType . . . 384
8.405packgetDyingGaspCfg Struct Reference . . . 384
8.405.1 Detailed Description . . . 384
8.405.2 Field Documentation . . . 384
8.405.2.1 pDestSMSContent . . . 384
8.405.2.2 pDestSMSNum . . . 384
8.406packgetDyingGaspStatistics Struct Reference . . . 384
8.406.1 Detailed Description . . . 384
8.406.2 Field Documentation . . . 385
8.406.2.1 pSMSAttemptedFlag . . . 385
8.406.2.2 pTimeStamp . . . 385
8.407qmiSmsMessageList Struct Reference . . . 385
8.407.1 Detailed Description . . . 385
8.407.2 Field Documentation . . . 385
8.407.2.1 messageIndex . . . 385
8.407.2.2 messageTag . . . 385
8.408qmiWDSDataBearerTechnology Struct Reference . . . 385
8.408.1 Detailed Description . . . 385
8.408.2 Field Documentation . . . 386
8.408.2.1 currentNetwork . . . 386
8.408.2.2 ratMask . . . 386
8.408.2.3 soMask . . . 386
8.409RFBandInfoElements Struct Reference . . . 386
8.409.1 Detailed Description . . . 387
8.409.2 Field Documentation . . . 387
8.409.2.1 activeBandClass . . . 387
8.409.2.2 activeChannel . . . 387
8.409.2.3 radioInterface . . . 387
8.410rmTrasnferStaticsReq Struct Reference . . . 387
8.410.1 Detailed Description . . . 387
8.410.2 Field Documentation . . . 388
8.410.2.1 bResetStatistics . . . 388
8.410.2.2 ulMask . . . 388
8.411sensorData_t Struct Reference . . . 388
8.411.1 Detailed Description . . . 388
8.411.2 Field Documentation . . . 389

8.411.2.1 flags . . . 389
8.411.2.2 sensorDataLen . . . 389
8.411.2.3 timeOfFirstSample . . . 389
8.411.2.4 timeOffset . . . 389
8.411.2.5 xAxis . . . 389
8.411.2.6 yAxis . . . 389
8.411.2.7 zAxis . . . 389
8.412slot_t Struct Reference . . . 390
8.412.1 Detailed Description . . . 390
8.412.2 Field Documentation . . . 390
8.412.2.1 bICCID . . . 390
8.412.2.2 bICCIDLength . . . 390
8.412.2.3 bLogicalSlot . . . 390
8.412.2.4 uPhyCardStatus . . . 390
8.412.2.5 uPhySlotStatus . . . 391
8.413slotInf Struct Reference . . . 391
8.413.1 Detailed Description . . . 391
8.413.2 Field Documentation . . . 392
8.413.2.1 AppStatus . . . 392
8.413.2.2 cardState . . . 392
8.413.2.3 errorState . . . 392
8.413.2.4 numApp . . . 392
8.413.2.5 upinRetries . . . 392
8.413.2.6 upinState . . . 392
8.413.2.7 upukRetries . . . 392
8.414slots_t Struct Reference . . . 392
8.414.1 Field Documentation . . . 392
8.414.1.1 uimSlotStatus . . . 392
8.415sms_BroadcastConfig Struct Reference . . . 392
8.415.1 Detailed Description . . . 393
8.415.2 Field Documentation . . . 393
8.415.2.1 fromServiceId . . . 393
8.415.2.2 selected . . . 393
8.415.2.3 toServiceId . . . 393
8.416sms_CDMABroadcastConfig Struct Reference . . . 393
8.416.1 Detailed Description . . . 393
8.416.2 Field Documentation . . . 394
8.416.2.1 language . . . 394
8.416.2.2 selected . . . 394
8.416.2.3 serviceCategory . . . 394

8.417 sms_getIndicationReg Struct Reference . . . 394
8.417.1 Detailed Description . . . 394
8.417.2 Field Documentation . . . 395
8.417.2.1 pRegCallStatInfoEvt . . . 395
8.417.2.2 pRegTransLayerInfoEvt . . . 395
8.417.2.3 pRegTransNWRegInfoEvt . . . 395
8.418 sms_getMsgWaitingInfo Struct Reference . . . 395
8.418.1 Detailed Description . . . 395
8.418.2 Field Documentation . . . 395
8.418.2.1 msgWaitInfo . . . 395
8.418.2.2 numInstances . . . 395
8.419 sms_getTransLayerInfo Struct Reference . . . 395
8.419.1 Detailed Description . . . 396
8.419.2 Field Documentation . . . 396
8.419.2.1 pRegInd . . . 396
8.419.2.2 pTransLayerInfo . . . 396
8.420 sms_getTransNWRegInfo Struct Reference . . . 396
8.420.1 Detailed Description . . . 396
8.420.2 Field Documentation . . . 396
8.420.2.1 pRegStatus . . . 397
8.421 sms_maxStorageSizeReq Struct Reference . . . 397
8.421.1 Detailed Description . . . 397
8.421.2 Field Documentation . . . 397
8.421.2.1 pMessageMode . . . 397
8.421.2.2 storageType . . . 397
8.422 sms_maxStorageSizeResp Struct Reference . . . 397
8.422.1 Detailed Description . . . 397
8.422.2 Field Documentation . . . 398
8.422.2.1 freeSlots . . . 398
8.422.2.2 maxStorageSize . . . 398
8.423 sms_messageWaitingInfoContent Struct Reference . . . 398
8.423.1 Detailed Description . . . 398
8.423.2 Field Documentation . . . 398
8.423.2.1 activeInd . . . 398
8.423.2.2 msgCount . . . 398
8.423.2.3 msgType . . . 398
8.424 sms_msgProtocolResp Struct Reference . . . 399
8.424.1 Detailed Description . . . 399
8.424.2 Field Documentation . . . 399
8.424.2.1 msgProtocol . . . 399

8.425 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference . . . 399
8.425.1 Detailed Description . . . 399
8.425.2 Field Documentation . . . 400
8.425.2.1 activated_ind . . . 400
8.425.2.2 CDMABroadcastConfig . . . 400
8.425.2.3 num_instances . . . 400
8.426 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference . . . 400
8.426.1 Detailed Description . . . 400
8.426.2 Field Documentation . . . 400
8.426.2.1 activated_ind . . . 400
8.426.2.2 broadcastConfig . . . 400
8.426.2.3 num_instances . . . 400
8.427 sms_routeEntry Struct Reference . . . 400
8.427.1 Detailed Description . . . 401
8.427.2 Field Documentation . . . 401
8.427.2.1 messageClass . . . 401
8.427.2.2 messageType . . . 401
8.427.2.3 receiptAction . . . 401
8.427.2.4 routeStorage . . . 401
8.428 sms_sendAsyncsmsParams Struct Reference . . . 401
8.428.1 Detailed Description . . . 402
8.428.2 Field Documentation . . . 403
8.428.2.1 messageFormat . . . 403
8.428.2.2 messageSize . . . 403
8.428.2.3 pFollowOnDC . . . 403
8.428.2.4 pForceOnDC . . . 403
8.428.2.5 pLinktimer . . . 403
8.428.2.6 pMessage . . . 403
8.428.2.7 pRetryMessage . . . 403
8.428.2.8 pRetryMessageId . . . 403
8.428.2.9 pServiceOption . . . 403
8.428.2.10 pSmsOnIms . . . 403
8.428.2.11 pUserData . . . 403
8.429 sms_setIndicationReg Struct Reference . . . 404
8.429.1 Detailed Description . . . 404
8.429.2 Field Documentation . . . 404
8.429.2.1 pRegCallStatInfoEvt . . . 404
8.429.2.2 pRegTransLayerInfoEvt . . . 404
8.429.2.3 pRegTransNWRegInfoEvt . . . 404
8.430 sms_setRoutesReq Struct Reference . . . 404

8.430.1 Detailed Description . . . 405
8.430.2 Field Documentation . . . 405
8.430.2.1 numOfRoutes . . . 405
8.430.2.2 pTransferStatusReport . . . 405
8.430.2.3 routeList . . . 405
8.431 sms_transLayerInfo Struct Reference . . . 405
8.431.1 Detailed Description . . . 405
8.431.2 Field Documentation . . . 405
8.431.2.1 TransCap . . . 406
8.431.2.2 TransType . . . 406
8.432 sMSCAddress Struct Reference . . . 406
8.432.1 Detailed Description . . . 406
8.432.2 Field Documentation . . . 406
8.432.2.1 data . . . 406
8.432.2.2 length . . . 406
8.433 sMSCAddressTlv Struct Reference . . . 406
8.433.1 Detailed Description . . . 406
8.433.2 Field Documentation . . . 406
8.433.2.1 SMSCInfo . . . 406
8.433.2.2 TlvPresent . . . 406
8.434 sMSEtwsMessage Struct Reference . . . 407
8.434.1 Detailed Description . . . 407
8.434.2 Field Documentation . . . 407
8.434.2.1 data . . . 407
8.434.2.2 length . . . 407
8.434.2.3 notificationType . . . 407
8.435 sMSEtwsMessageTlv Struct Reference . . . 407
8.435.1 Detailed Description . . . 407
8.435.2 Field Documentation . . . 407
8.435.2.1 EtwsMessageInfo . . . 407
8.435.2.2 TlvPresent . . . 408
8.436 sMSEtwsPlmn Struct Reference . . . 408
8.436.1 Detailed Description . . . 408
8.436.2 Field Documentation . . . 408
8.436.2.1 mobileCountryCode . . . 408
8.436.2.2 mobileNetworkCode . . . 408
8.437 sMSMessageMode Struct Reference . . . 408
8.437.1 Detailed Description . . . 408
8.437.2 Field Documentation . . . 408
8.437.2.1 messageMode . . . 408

8.438 sMSMTMessage Struct Reference . . . 408
8.438.1 Detailed Description . . . 408
8.438.2 Field Documentation . . . 409
8.438.2.1 messageIndex . . . 409
8.438.2.2 storageType . . . 409
8.439 sMSOnIMS Struct Reference . . . 409
8.439.1 Detailed Description . . . 409
8.439.2 Field Documentation . . . 409
8.439.2.1 smsOnIMS . . . 409
8.440 sMSOnIMSTlv Struct Reference . . . 409
8.440.1 Detailed Description . . . 409
8.440.2 Field Documentation . . . 409
8.440.2.1 IMSInfo . . . 409
8.440.2.2 TlvPresent . . . 409
8.441 sMSTransferRouteMTMessage Struct Reference . . . 410
8.441.1 Detailed Description . . . 410
8.441.2 Field Documentation . . . 410
8.441.2.1 ackIndicator . . . 410
8.441.2.2 data . . . 410
8.441.2.3 format . . . 410
8.441.2.4 length . . . 410
8.441.2.5 transactionID . . . 410
8.442 tdscdmaSigInfoExt Struct Reference . . . 410
8.442.1 Detailed Description . . . 411
8.442.2 Field Documentation . . . 411
8.442.2.1 ecio . . . 411
8.442.2.2 rscp . . . 411
8.442.2.3 rssi . . . 411
8.442.2.4 sinr . . . 411
8.443 tempData_t Struct Reference . . . 411
8.443.1 Detailed Description . . . 411
8.443.2 Field Documentation . . . 412
8.443.2.1 temperature . . . 412
8.443.2.2 temperatureDataLen . . . 412
8.443.2.3 timeOfFirstSample . . . 412
8.443.2.4 timeOffset . . . 412
8.443.2.5 timeSource . . . 412
8.444 transferRouteMessageTlv Struct Reference . . . 412
8.444.1 Detailed Description . . . 412
8.444.2 Field Documentation . . . 413

8.444.2.1 TlvPresent . . . 413
8.444.2.2 TransferRouteMTMessageInfo . . . 413
8.445 transferStatInd Struct Reference . . . 413
8.445.1 Detailed Description . . . 413
8.445.2 Field Documentation . . . 413
8.445.2.1 StatsMask . . . 413
8.445.2.2 StatsPeriod . . . 413
8.446 uim_appStatus Struct Reference . . . 413
8.446.1 Detailed Description . . . 413
8.446.2 Field Documentation . . . 415
8.446.2.1 aidLength . . . 415
8.446.2.2 aidVal . . . 415
8.446.2.3 appState . . . 416
8.446.2.4 appType . . . 416
8.446.2.5 persoFeature . . . 416
8.446.2.6 persoRetries . . . 416
8.446.2.7 persoState . . . 416
8.446.2.8 persoUnblockRetries . . . 416
8.446.2.9 pin1Retries . . . 416
8.446.2.10 pin1State . . . 416
8.446.2.11 pin2Retries . . . 416
8.446.2.12 pin2State . . . 416
8.446.2.13 puk1Retries . . . 416
8.446.2.14 puk2Retries . . . 416
8.446.2.15 univPin . . . 416
8.447 uim_authenticateResult Struct Reference . . . 416
8.447.1 Detailed Description . . . 416
8.447.2 Field Documentation . . . 416
8.447.2.1 content . . . 416
8.447.2.2 contentLen . . . 416
8.448 uim_authenticationData Struct Reference . . . 417
8.448.1 Detailed Description . . . 417
8.448.2 Field Documentation . . . 417
8.448.2.1 context . . . 417
8.448.2.2 data . . . 417
8.448.2.3 dataLen . . . 417
8.449 uim_cardResult Struct Reference . . . 418
8.449.1 Detailed Description . . . 418
8.449.2 Field Documentation . . . 418
8.449.2.1 sw1 . . . 418

8.449.2.2 sw2 . . . 418
8.450 uim_cardStatus Struct Reference . . . 418
8.450.1 Detailed Description . . . 418
8.450.2 Field Documentation . . . 419
8.450.2.1 index1xPri . . . 419
8.450.2.2 index1xSec . . . 419
8.450.2.3 indexGwPri . . . 419
8.450.2.4 indexGwSec . . . 419
8.450.2.5 numSlot . . . 419
8.450.2.6 SlotInfo . . . 419
8.451 uim_changeUIMPIN Struct Reference . . . 419
8.451.1 Detailed Description . . . 420
8.451.2 Field Documentation . . . 420
8.451.2.1 oldPINLen . . . 420
8.451.2.2 oldPINVal . . . 420
8.451.2.3 pinID . . . 420
8.451.2.4 pinLen . . . 420
8.451.2.5 pinVal . . . 420
8.452 uim_depersonalizationInformation Struct Reference . . . 420
8.452.1 Detailed Description . . . 420
8.452.2 Field Documentation . . . 421
8.452.2.1 ckLen . . . 421
8.452.2.2 ckVal . . . 421
8.452.2.3 feature . . . 421
8.452.2.4 operation . . . 421
8.453 uim_encryptedPIN1 Struct Reference . . . 421
8.453.1 Detailed Description . . . 421
8.453.2 Field Documentation . . . 422
8.453.2.1 pin1Len . . . 422
8.453.2.2 pin1Val . . . 422
8.454 uim_fileAttributes Struct Reference . . . 422
8.454.1 Detailed Description . . . 422
8.454.2 Field Documentation . . . 425
8.454.2.1 fileID . . . 425
8.454.2.2 fileSize . . . 425
8.454.2.3 fileType . . . 425
8.454.2.4 rawLen . . . 425
8.454.2.5 rawValue . . . 425
8.454.2.6 recordCount . . . 425
8.454.2.7 recordSize . . . 425

8.454.2.8 secActivate . . . 425
8.454.2.9 secActivateMask . . . 425
8.454.2.10 secDeactivate . . . 425
8.454.2.11 secDeactivateMask . . . 425
8.454.2.12 secIncrease . . . 425
8.454.2.13 secIncreaseMask . . . 425
8.454.2.14 secRead . . . 425
8.454.2.15 secReadMask . . . 425
8.454.2.16 secWrite . . . 425
8.454.2.17 secWriteMask . . . 425
8.455 uim_fileInfo Struct Reference . . . 425
8.455.1 Detailed Description . . . 425
8.455.2 Field Documentation . . . 426
8.455.2.1 fileID . . . 426
8.455.2.2 path . . . 426
8.455.2.3 pathLen . . . 426
8.456 uim_hotSwapStatus Struct Reference . . . 426
8.456.1 Detailed Description . . . 426
8.456.2 Field Documentation . . . 426
8.456.2.1 hotSwap . . . 426
8.456.2.2 hotSwapLength . . . 426
8.457 uim_personalizationStatus Struct Reference . . . 426
8.457.1 Detailed Description . . . 427
8.457.2 Field Documentation . . . 427
8.457.2.1 feature . . . 427
8.457.2.2 numFeatures . . . 427
8.457.2.3 unblockLeft . . . 427
8.457.2.4 verifyLeft . . . 427
8.458 uim_readResult Struct Reference . . . 427
8.458.1 Detailed Description . . . 428
8.458.2 Field Documentation . . . 428
8.458.2.1 content . . . 428
8.458.2.2 contentLen . . . 428
8.459 uim_readTransparentInfo Struct Reference . . . 428
8.459.1 Detailed Description . . . 428
8.459.2 Field Documentation . . . 428
8.459.2.1 length . . . 428
8.459.2.2 offset . . . 428
8.460 uim_refreshevent Struct Reference . . . 428
8.460.1 Detailed Description . . . 429

8.460.2 Field Documentation . . . 430
8.460.2.1 aid . . . 430
8.460.2.2 aidLength . . . 430
8.460.2.3 arrfileInfo . . . 430
8.460.2.4 mode . . . 430
8.460.2.5 numOfFiles . . . 430
8.460.2.6 sessionType . . . 430
8.460.2.7 stage . . . 430
8.461 uim_registerRefresh Struct Reference . . . 430
8.461.1 Detailed Description . . . 430
8.461.2 Field Documentation . . . 431
8.461.2.1 arrfileInfo . . . 431
8.461.2.2 numFiles . . . 431
8.461.2.3 registerFlag . . . 431
8.461.2.4 voteForInit . . . 431
8.462 uim_remainingRetries Struct Reference . . . 431
8.462.1 Detailed Description . . . 431
8.462.2 Field Documentation . . . 431
8.462.2.1 unblockLeft . . . 431
8.462.2.2 verifyLeft . . . 432
8.463 uim_sessionInformation Struct Reference . . . 432
8.463.1 Detailed Description . . . 432
8.463.2 Field Documentation . . . 432
8.463.2.1 aid . . . 432
8.463.2.2 aidLength . . . 432
8.463.2.3 sessionType . . . 432
8.464 uim_setPINProtection Struct Reference . . . 432
8.464.1 Detailed Description . . . 433
8.464.2 Field Documentation . . . 433
8.464.2.1 pinID . . . 433
8.464.2.2 pinLength . . . 433
8.464.2.3 pinOperation . . . 433
8.464.2.4 pinValue . . . 433
8.465 uim_slotInfo Struct Reference . . . 433
8.465.1 Detailed Description . . . 434
8.465.2 Field Documentation . . . 435
8.465.2.1 AppStatus . . . 435
8.465.2.2 cardState . . . 435
8.465.2.3 errorState . . . 435
8.465.2.4 numApp . . . 435

8.465.2.5 upinRetries . . . 435
8.465.2.6 upinState . . . 435
8.465.2.7 upukRetries . . . 435
8.466 uim_UIMSessionInformation Struct Reference . . . 435
8.466.1 Detailed Description . . . 435
8.466.2 Field Documentation . . . 436
8.466.2.1 aid . . . 436
8.466.2.2 aidLength . . . 436
8.466.2.3 sessionType . . . 436
8.467 uim_unblockUIMPIN Struct Reference . . . 436
8.467.1 Detailed Description . . . 436
8.467.2 Field Documentation . . . 436
8.467.2.1 newPINLen . . . 436
8.467.2.2 newPINVal . . . 436
8.467.2.3 pinID . . . 436
8.467.2.4 pukLen . . . 437
8.467.2.5 pukVal . . . 437
8.468 uim_verifyUIMPIN Struct Reference . . . 437
8.468.1 Detailed Description . . . 437
8.468.2 Field Documentation . . . 437
8.468.2.1 pinID . . . 437
8.468.2.2 pinLen . . . 437
8.468.2.3 pinVal . . . 437
8.469 unpack_cat_SetCatEventCallback_ind_t Struct Reference . . . 437
8.469.1 Detailed Description . . . 437
8.469.2 Field Documentation . . . 438
8.469.2.1 CCETlv . . . 438
8.469.2.2 event_Index . . . 438
8.470 unpack_cat_SetCATEventCallback_t Struct Reference . . . 438
8.470.1 Detailed Description . . . 438
8.470.2 Field Documentation . . . 438
8.470.2.1 errorMask . . . 438
8.470.2.2 Tlvresult . . . 438
8.471 unpack_dms_GetActivationState_t Struct Reference . . . 438
8.471.1 Detailed Description . . . 439
8.471.2 Field Documentation . . . 439
8.471.2.1 state . . . 439
8.472 unpack_dms_GetBandCapability_t Struct Reference . . . 439
8.472.1 Detailed Description . . . 439
8.472.2 Field Documentation . . . 440

8.472.2.1 BandCapability . . . 441
8.472.2.2 Tlvresult . . . 441
8.473 unpack_dms_GetCrashAction_t Struct Reference . . . 441
8.473.1 Detailed Description . . . 441
8.473.2 Field Documentation . . . 441
8.473.2.1 DevCrashState . . . 441
8.473.2.2 Tlvresult . . . 441
8.474 unpack_dms_GetCustFeature_t Struct Reference . . . 441
8.474.1 Detailed Description . . . 442
8.474.2 Field Documentation . . . 443
8.474.2.1 DHCPRelayEnabled . . . 443
8.474.2.2 DisableIMSI . . . 443
8.474.2.3 GpsEnable . . . 443
8.474.2.4 GPSLPM . . . 443
8.474.2.5 GPSSel . . . 443
8.474.2.6 IPFamSupport . . . 443
8.474.2.7 IsVoiceEnabled . . . 443
8.474.2.8 RMAutoConnect . . . 443
8.474.2.9 SMSSupport . . . 443
8.474.2.10 Tlvresult . . . 443
8.475 unpack_dms_GetCustFeaturesV2_t Struct Reference . . . 443
8.475.1 Detailed Description . . . 444
8.475.2 Field Documentation . . . 444
8.475.2.1 GetCustomFeatureV2 . . . 444
8.475.2.2 Tlvresult . . . 444
8.476 unpack_dms_GetDeviceCap_t Struct Reference . . . 444
8.476.1 Detailed Description . . . 444
8.476.2 Field Documentation . . . 445
8.476.2.1 DataServiceCapability . . . 445
8.476.2.2 MaxRXChannelRate . . . 445
8.476.2.3 MaxTXChannelRate . . . 445
8.476.2.4 RadioIfaces . . . 445
8.476.2.5 RadioIfacesSize . . . 445
8.476.2.6 SimCapability . . . 445
8.476.2.7 Tlvresult . . . 445
8.477 unpack_dms_GetDeviceCapabilities_t Struct Reference . . . 445
8.477.1 Detailed Description . . . 445
8.477.2 Field Documentation . . . 446
8.477.2.1 dataServiceCaCapability . . . 446
8.477.2.2 maxRxChannelRate . . . 446

8.477.2.3 maxTxChannelRate . . . 446
8.477.2.4 RadioIfaces . . . 446
8.477.2.5 radioIfacesSize . . . 446
8.477.2.6 simCapability . . . 446
8.478 unpack_dms_GetDeviceHardwareRev_t Struct Reference . . . 446
8.478.1 Detailed Description . . . 447
8.478.2 Field Documentation . . . 447
8.478.2.1 String . . . 447
8.478.2.2 stringSize . . . 447
8.478.2.3 Tlvresult . . . 447
8.479 unpack_dms_GetDeviceMfr_t Struct Reference . . . 447
8.479.1 Detailed Description . . . 447
8.479.2 Field Documentation . . . 447
8.479.2.1 String . . . 448
8.479.2.2 stringSize . . . 448
8.479.2.3 Tlvresult . . . 448
8.480 unpack_dms_GetDeviceSerialNumbers_t Struct Reference . . . 448
8.480.1 Detailed Description . . . 448
8.480.2 Field Documentation . . . 449
8.480.2.1 esnSize . . . 449
8.480.2.2 ESNString . . . 449
8.480.2.3 imeiSize . . . 449
8.480.2.4 IMEIString . . . 449
8.480.2.5 imeiSvnSize . . . 449
8.480.2.6 ImeiSvnString . . . 449
8.480.2.7 meidSize . . . 449
8.480.2.8 MEIDString . . . 449
8.480.2.9 Tlvresult . . . 449
8.481 unpack_dms_GetFirmwareInfo_t Struct Reference . . . 449
8.481.1 Detailed Description . . . 449
8.481.2 Field Documentation . . . 450
8.481.2.1 appversion_str . . . 450
8.481.2.2 bootversion_str . . . 450
8.481.2.3 carrier_str . . . 450
8.481.2.4 cur_carr_name . . . 450
8.481.2.5 cur_carr_rev . . . 450
8.481.2.6 modelid_str . . . 450
8.481.2.7 packageid_str . . . 450
8.481.2.8 priversion_str . . . 450
8.481.2.9 sku_str . . . 450

8.481.2.10 Tlvresult . . . 450
8.482 unpack_dms_GetFirmwareRevision_t Struct Reference . . . 451
8.482.1 Detailed Description . . . 451
8.482.2 Field Documentation . . . 451
8.482.2.1 amssSize . . . 451
8.482.2.2 AMSSString . . . 451
8.482.2.3 PRIString . . . 451
8.482.2.4 Tlvresult . . . 451
8.483 unpack_dms_GetFirmwareRevisions_t Struct Reference . . . 451
8.483.1 Detailed Description . . . 452
8.483.2 Field Documentation . . . 452
8.483.2.1 amssSize . . . 452
8.483.2.2 AMSSString . . . 452
8.483.2.3 bootSize . . . 452
8.483.2.4 BootString . . . 452
8.483.2.5 priSize . . . 452
8.483.2.6 PRIString . . . 452
8.483.2.7 Tlvresult . . . 452
8.484 unpack_dms_GetFSN_t Struct Reference . . . 452
8.484.1 Detailed Description . . . 453
8.484.2 Field Documentation . . . 453
8.484.2.1 String . . . 453
8.484.2.2 Tlvresult . . . 453
8.485 unpack_dms_GetHardwareRevision_t Struct Reference . . . 453
8.485.1 Detailed Description . . . 453
8.485.2 Field Documentation . . . 453
8.485.2.1 hwVer . . . 453
8.486 unpack_dms_GetIMSI_t Struct Reference . . . 453
8.486.1 Detailed Description . . . 454
8.486.2 Field Documentation . . . 454
8.486.2.1 imsi . . . 454
8.486.2.2 Tlvresult . . . 454
8.487 unpack_dms_GetManufacturer_t Struct Reference . . . 454
8.487.1 Detailed Description . . . 454
8.487.2 Field Documentation . . . 454
8.487.2.1 manufacturer . . . 454
8.487.2.2 Tlvresult . . . 454
8.488 unpack_dms_GetModelID_t Struct Reference . . . 454
8.488.1 Detailed Description . . . 454
8.488.2 Field Documentation . . . 455

8.488.2.1 modelid . . . 455
8.488.2.2 Tlvresult . . . 455
8.489 unpack_dms_GetNetworkTime_t Struct Reference . . . 455
8.489.1 Detailed Description . . . 455
8.489.2 Field Documentation . . . 455
8.489.2.1 source . . . 455
8.489.2.2 timestamp . . . 455
8.489.2.3 Tlvresult . . . 455
8.490 unpack_dms_GetOfflineReason_t Struct Reference . . . 456
8.490.1 Detailed Description . . . 456
8.490.2 Field Documentation . . . 456
8.490.2.1 pbPlatform . . . 456
8.490.2.2 pReasonMask . . . 456
8.490.2.3 Tlvresult . . . 456
8.491 unpack_dms_GetPower_t Struct Reference . . . 456
8.491.1 Detailed Description . . . 457
8.491.2 Field Documentation . . . 457
8.491.2.1 HardwareControlledMode . . . 457
8.491.2.2 OfflineReason . . . 457
8.491.2.3 OperationMode . . . 457
8.491.2.4 Tlvresult . . . 457
8.492 unpack_dms_GetPRLVersion_t Struct Reference . . . 457
8.492.1 Detailed Description . . . 458
8.492.2 Field Documentation . . . 458
8.492.2.1 Tlvresult . . . 458
8.492.2.2 u16PRLVersion . . . 458
8.492.2.3 u8PRLPreference . . . 458
8.493 unpack_dms_GetSerialNumbers_t Struct Reference . . . 458
8.493.1 Detailed Description . . . 458
8.493.2 Field Documentation . . . 459
8.493.2.1 esn . . . 459
8.493.2.2 imei_no . . . 459
8.493.2.3 imeisv_svn . . . 459
8.493.2.4 meid . . . 459
8.494 unpack_dms_GetUSBComp_t Struct Reference . . . 459
8.494.1 Detailed Description . . . 459
8.494.2 Field Documentation . . . 460
8.494.2.1 NumSupUSBComps . . . 460
8.494.2.2 SupUSBComps . . . 460
8.494.2.3 Tlvresult . . . 460

8.494.2.4 USBComp . . . 460
8.495 unpack_dms_GetVoiceNumber_t Struct Reference . . . 460
8.495.1 Detailed Description . . . 461
8.495.2 Field Documentation . . . 461
8.495.2.1 MIN . . . 461
8.495.2.2 minSize . . . 461
8.495.2.3 Tlvresult . . . 461
8.495.2.4 VoiceNumber . . . 461
8.495.2.5 voiceNumberSize . . . 461
8.496 unpack_dms_ResetToFactoryDefaults_t Struct Reference . . . 461
8.496.1 Detailed Description . . . 461
8.496.2 Field Documentation . . . 462
8.496.2.1 Tlvresult . . . 462
8.497 unpack_dms_SetActivationStatusCallback_t Struct Reference . . . 462
8.497.1 Detailed Description . . . 462
8.497.2 Field Documentation . . . 462
8.497.2.1 Tlvresult . . . 462
8.498 unpack_dms_SetCrashAction_t Struct Reference . . . 462
8.498.1 Detailed Description . . . 462
8.498.2 Field Documentation . . . 462
8.498.2.1 notused . . . 462
8.499 unpack_dms_SetCustFeature_t Struct Reference . . . 462
8.499.1 Detailed Description . . . 463
8.499.2 Field Documentation . . . 463
8.499.2.1 Tlvresult . . . 463
8.500 unpack_dms_SetCustFeaturesV2_t Struct Reference . . . 463
8.500.1 Detailed Description . . . 463
8.500.2 Field Documentation . . . 463
8.500.2.1 Tlvresult . . . 463
8.501 unpack_dms_SetEventReport_ind_t Struct Reference . . . 463
8.501.1 Detailed Description . . . 463
8.501.2 Field Documentation . . . 464
8.501.2.1 ActivationStatusTlv . . . 464
8.501.2.2 OperatingModeTlv . . . 464
8.501.2.3 Tlvresult . . . 464
8.502 unpack_dms_SetEventReport_t Struct Reference . . . 464
8.502.1 Detailed Description . . . 464
8.502.2 Field Documentation . . . 464
8.502.2.1 Tlvresult . . . 464
8.503 unpack_dms_SetFirmwarePreference_t Struct Reference . . . 464

8.503.1 Detailed Description . . . 464
8.503.2 Field Documentation . . . 464
8.503.2.1 TlvResult . . . 465
8.504 unpack_dms_SetPower_t Struct Reference . . . 465
8.504.1 Detailed Description . . . 465
8.504.2 Field Documentation . . . 465
8.504.2.1 TlvResult . . . 465
8.505 unpack_dms_SetUSBComp_t Struct Reference . . . 465
8.505.1 Detailed Description . . . 465
8.505.2 Field Documentation . . . 465
8.505.2.1 TlvResult . . . 465
8.506 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference . . . 465
8.506.1 Detailed Description . . . 466
8.506.2 Field Documentation . . . 466
8.506.2.1 source . . . 466
8.506.2.2 TlvResult . . . 466
8.506.2.3 type . . . 466
8.507 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference . . . 466
8.507.1 Detailed Description . . . 466
8.507.2 Field Documentation . . . 467
8.507.2.1 source . . . 467
8.507.2.2 TlvResult . . . 467
8.507.2.3 type . . . 467
8.508 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 467
8.508.1 Detailed Description . . . 467
8.508.2 Field Documentation . . . 468
8.508.2.1 TlvResult . . . 468
8.509 unpack_dms_SLQSGetBandCapability_t Struct Reference . . . 468
8.509.1 Detailed Description . . . 468
8.509.2 Field Documentation . . . 471
8.509.2.1 bandCapability . . . 471
8.509.2.2 is_LteBandCapability_Available . . . 471
8.509.2.3 is_TdsBandCapability_Available . . . 471
8.509.2.4 LteBandCapability . . . 471
8.509.2.5 TdsBandCapability . . . 471
8.510 unpack_dms_SLQSGetERIFile_t Struct Reference . . . 471
8.510.1 Detailed Description . . . 471
8.510.2 Field Documentation . . . 471
8.510.2.1 eriFile . . . 471
8.510.2.2 TlvResult . . . 471

8.511 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference . . . 471
8.511.1 Detailed Description . . . 472
8.511.2 Field Documentation . . . 472
8.511.2.1 TlvResult . . . 472
8.512 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 472
8.512.1 Detailed Description . . . 472
8.512.2 Field Documentation . . . 472
8.512.2.1 crashInfoParam . . . 472
8.512.2.2 TlvResult . . . 472
8.513 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference . . . 472
8.513.1 Detailed Description . . . 473
8.513.2 Field Documentation . . . 473
8.513.2.1 pGetDyingGaspCfg . . . 473
8.513.2.2 TlvResult . . . 473
8.514 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference . . . 473
8.514.1 Detailed Description . . . 473
8.514.2 Field Documentation . . . 473
8.514.2.1 pGetDyingGaspStatistics . . . 473
8.514.2.2 TlvResult . . . 473
8.515 unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference . . . 473
8.515.1 Detailed Description . . . 474
8.515.2 Field Documentation . . . 474
8.515.2.1 carrier . . . 474
8.515.2.2 fwvers . . . 474
8.515.2.3 numEntries . . . 474
8.515.2.4 pCurrImgInfo . . . 474
8.515.2.5 pkgver . . . 474
8.515.2.6 priver . . . 474
8.516 unpack_dms_SLQSSwiGetFwUpdateStatus_t Struct Reference . . . 474
8.516.1 Detailed Description . . . 475
8.516.2 Field Documentation . . . 475
8.516.2.1 imgType . . . 475
8.516.2.2 logString . . . 475
8.516.2.3 refData . . . 475
8.516.2.4 refString . . . 476
8.516.2.5 ResCode . . . 476
8.516.2.6 TlvResult . . . 476
8.517 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference . . . 476
8.517.1 Detailed Description . . . 476
8.517.2 Field Documentation . . . 476

8.517.2.1 manString . . . 476
8.517.2.2 modelString . . . 476
8.517.2.3 plasmaIDString . . . 476
8.517.2.4 swVerString . . . 476
8.517.2.5 TlvResult . . . 476
8.518 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference . . . 477
8.518.1 Detailed Description . . . 477
8.518.2 Field Documentation . . . 477
8.518.2.1 nameString . . . 477
8.518.2.2 TlvResult . . . 477
8.518.2.3 versionString . . . 477
8.519 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference . . . 477
8.519.1 Detailed Description . . . 477
8.519.2 Field Documentation . . . 478
8.519.2.1 meidString . . . 478
8.519.2.2 TlvResult . . . 478
8.520 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 478
8.520.1 Detailed Description . . . 478
8.520.2 Field Documentation . . . 478
8.520.2.1 TlvResult . . . 478
8.521 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 478
8.521.1 Detailed Description . . . 478
8.521.2 Field Documentation . . . 478
8.521.2.1 TlvResult . . . 478
8.522 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 478
8.522.1 Detailed Description . . . 479
8.522.2 Field Documentation . . . 479
8.522.2.1 TlvResult . . . 479
8.523 unpack_dms_SLQSUIMGetState_t Struct Reference . . . 479
8.523.1 Detailed Description . . . 479
8.523.2 Field Documentation . . . 479
8.523.2.1 state . . . 479
8.523.2.2 TlvResult . . . 479
8.524 unpack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 479
8.524.1 Detailed Description . . . 479
8.524.2 Field Documentation . . . 480
8.524.2.1 facilityState . . . 480
8.524.2.2 TlvResult . . . 480
8.524.2.3 unblockRetriesLeft . . . 480
8.524.2.4 verifyRetriesLeft . . . 480

8.525 unpack_dms_UIMGetICCID_t Struct Reference . . . 480
8.525.1 Detailed Description . . . 480
8.525.2 Field Documentation . . . 481
8.525.2.1 String . . . 481
8.525.2.2 stringSize . . . 481
8.525.2.3 Tlvresult . . . 481
8.526 unpack_dms_UIMGetPINStatus_t Struct Reference . . . 481
8.526.1 Detailed Description . . . 481
8.526.2 Field Documentation . . . 482
8.526.2.1 p1Status . . . 482
8.526.2.2 p1UnblockRetriesLeft . . . 482
8.526.2.3 p1VerifyRetriesLeft . . . 482
8.526.2.4 p2Status . . . 482
8.526.2.5 p2UnblockRetriesLeft . . . 482
8.526.2.6 p2VerifyRetriesLeft . . . 482
8.526.2.7 Tlvresult . . . 482
8.527 unpack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 482
8.527.1 Detailed Description . . . 482
8.527.2 Field Documentation . . . 483
8.527.2.1 Tlvresult . . . 483
8.527.2.2 verifyRetriesLeft . . . 483
8.528 unpack_dms_UIMSetPINProtection_t Struct Reference . . . 483
8.528.1 Detailed Description . . . 483
8.528.2 Field Documentation . . . 483
8.528.2.1 Tlvresult . . . 483
8.528.2.2 unblockRetriesLeft . . . 483
8.528.2.3 verifyRetriesLeft . . . 483
8.529 unpack_dms_UIMUnblockControlKey_t Struct Reference . . . 484
8.529.1 Detailed Description . . . 484
8.529.2 Field Documentation . . . 484
8.529.2.1 Tlvresult . . . 484
8.529.2.2 unblockRetriesLeft . . . 484
8.530 unpack_fms_GetImagesPreference_t Struct Reference . . . 484
8.530.1 Detailed Description . . . 484
8.530.2 Field Documentation . . . 485
8.530.2.1 ImageListSize . . . 485
8.530.2.2 pImageList . . . 485
8.530.2.3 Tlvresult . . . 485
8.531 unpack_fms_GetStoredImages_t Struct Reference . . . 485
8.531.1 Detailed Description . . . 485

8.531.2 Field Documentation . . . 485
8.531.2.1 imageList . . . 485
8.531.2.2 imagelistSize . . . 485
8.531.2.3 Tlvresult . . . 485
8.532 unpack_fms_SetImagesPreference_t Struct Reference . . . 485
8.532.1 Detailed Description . . . 485
8.532.2 Field Documentation . . . 486
8.532.2.1 ImageTypes . . . 486
8.532.2.2 ImageTypesSize . . . 486
8.532.2.3 Tlvresult . . . 486
8.533 unpack_loc_BestAvailPos_Ind_t Struct Reference . . . 486
8.533.1 Detailed Description . . . 487
8.533.2 Field Documentation . . . 490
8.533.2.1 pAltitudeWrtEllipsoid . . . 490
8.533.2.2 pAltitudeWrtMeanSeaLevel . . . 490
8.533.2.3 pGpsTime . . . 490
8.533.2.4 pHeading . . . 490
8.533.2.5 pHeadingUnc . . . 490
8.533.2.6 pHorCirConf . . . 490
8.533.2.7 pHorEllpConf . . . 490
8.533.2.8 pHorReliability . . . 491
8.533.2.9 pHorUncCircular . . . 491
8.533.2.10 pHorUncEllipseOrientAzimuth . . . 491
8.533.2.11 pHorUncEllipseSemiMajor . . . 491
8.533.2.12 pHorUncEllipseSemiMinor . . . 491
8.533.2.13 pLatitude . . . 491
8.533.2.14 pLongitude . . . 491
8.533.2.15 pMagneticDeviation . . . 491
8.533.2.16 pPrecisionDilution . . . 491
8.533.2.17 pSensorDataUsage . . . 491
8.533.2.18 pSpeedHorizontal . . . 491
8.533.2.19 pSpeedUnc . . . 491
8.533.2.20 pSpeedVertical . . . 491
8.533.2.21 pSpeedVerticalUnc . . . 491
8.533.2.22 pSvUsedforFix . . . 491
8.533.2.23 pTechnologyMask . . . 491
8.533.2.24 pTimeSrc . . . 491
8.533.2.25 pTimestampUtc . . . 491
8.533.2.26 pTimeUnc . . . 491
8.533.2.27 pVertConfidence . . . 491

8.533.2.28 pVertReliability . . . 491
8.533.2.29 pVertUnc . . . 491
8.533.2.30 pXid . . . 491
8.533.2.31 status . . . 491
8.533.2.32 TlvResult . . . 491
8.534 unpack_loc_Delete_Assist_Data_t Struct Reference . . . 491
8.534.1 Detailed Description . . . 492
8.534.2 Field Documentation . . . 492
8.534.2.1 TlvResult . . . 492
8.535 unpack_loc_DeleteAssistData_Ind_t Struct Reference . . . 492
8.535.1 Detailed Description . . . 492
8.535.2 Field Documentation . . . 492
8.535.2.1 status . . . 493
8.535.2.2 TlvResult . . . 493
8.536 unpack_loc_EngineState_Ind_t Struct Reference . . . 493
8.536.1 Detailed Description . . . 493
8.536.2 Field Documentation . . . 493
8.536.2.1 engineState . . . 493
8.536.2.2 TlvResult . . . 493
8.537 unpack_loc_EventNMEA_Ind_t Struct Reference . . . 493
8.537.1 Detailed Description . . . 493
8.537.2 Field Documentation . . . 494
8.537.2.1 NMEAData . . . 494
8.537.2.2 TlvResult . . . 494
8.538 unpack_loc_EventRegister_t Struct Reference . . . 494
8.538.1 Detailed Description . . . 494
8.538.2 Field Documentation . . . 494
8.538.2.1 TlvResult . . . 494
8.539 unpack_loc_GetServer_Ind_t Struct Reference . . . 494
8.539.1 Detailed Description . . . 494
8.539.2 Field Documentation . . . 495
8.539.2.1 pIPv4AddrInfo . . . 495
8.539.2.2 pIPv6AddrInfo . . . 495
8.539.2.3 pURL . . . 495
8.539.2.4 serverStatus . . . 495
8.539.2.5 serverType . . . 495
8.539.2.6 TlvResult . . . 495
8.540 unpack_loc_GnssSvInfo_Ind_t Struct Reference . . . 496
8.540.1 Detailed Description . . . 496
8.540.2 Field Documentation . . . 496

8.540.2.1 altitudeAssumed . . . 496
8.540.2.2 pSatelliteInfo . . . 496
8.540.2.3 Tlvresult . . . 496
8.541 unpack_loc_PositionRpt_Ind_t Struct Reference . . . 496
8.541.1 Detailed Description . . . 497
8.541.2 Field Documentation . . . 501
8.541.2.1 pAltitudeAssumed . . . 501
8.541.2.2 pAltitudeWrtEllipsoid . . . 501
8.541.2.3 pAltitudeWrtMeanSeaLevel . . . 501
8.541.2.4 pFixId . . . 501
8.541.2.5 pGpsTime . . . 501
8.541.2.6 pHeading . . . 502
8.541.2.7 pHeadingUnc . . . 502
8.541.2.8 pHorConfidence . . . 502
8.541.2.9 pHorReliability . . . 502
8.541.2.10 pHorUncCircular . . . 502
8.541.2.11 pHorUncEllipseOrientAzimuth . . . 502
8.541.2.12 pHorUncEllipseSemiMajor . . . 502
8.541.2.13 pHorUncEllipseSemiMinor . . . 502
8.541.2.14 pLatitude . . . 502
8.541.2.15 pLeapSeconds . . . 502
8.541.2.16 pLongitude . . . 502
8.541.2.17 pMagneticDeviation . . . 502
8.541.2.18 pPrecisionDilution . . . 502
8.541.2.19 pSensorDataUsage . . . 502
8.541.2.20 pSpeedHorizontal . . . 502
8.541.2.21 pSpeedUnc . . . 502
8.541.2.22 pSpeedVertical . . . 502
8.541.2.23 pSvUsedforFix . . . 502
8.541.2.24 pTechnologyMask . . . 502
8.541.2.25 pTimeSrc . . . 502
8.541.2.26 pTimestampUtc . . . 502
8.541.2.27 pTimeUnc . . . 502
8.541.2.28 pVertConfidence . . . 502
8.541.2.29 pVertReliability . . . 502
8.541.2.30 pVertUnc . . . 502
8.541.2.31 sessionId . . . 502
8.541.2.32 sessionStatus . . . 502
8.541.2.33 Tlvresult . . . 502
8.542 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference . . . 503

8.542.1 Detailed Description . . . 503
8.542.2 Field Documentation . . . 503
8.542.2.1 status . . . 503
8.542.2.2 Tlvresult . . . 503
8.543 unpack_loc_SetExtPowerState_t Struct Reference . . . 503
8.543.1 Detailed Description . . . 503
8.543.2 Field Documentation . . . 504
8.543.2.1 Tlvresult . . . 504
8.544 unpack_loc_SetOperationMode_Ind_t Struct Reference . . . 504
8.544.1 Detailed Description . . . 504
8.544.2 Field Documentation . . . 504
8.544.2.1 status . . . 504
8.544.2.2 Tlvresult . . . 504
8.545 unpack_loc_SetOperationMode_t Struct Reference . . . 504
8.545.1 Detailed Description . . . 504
8.545.2 Field Documentation . . . 505
8.545.2.1 Tlvresult . . . 505
8.546 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 505
8.546.1 Detailed Description . . . 505
8.546.2 Field Documentation . . . 505
8.546.2.1 Tlvresult . . . 505
8.547 unpack_loc_Start_t Struct Reference . . . 505
8.547.1 Detailed Description . . . 505
8.547.2 Field Documentation . . . 505
8.547.2.1 Tlvresult . . . 505
8.548 unpack_loc_Stop_t Struct Reference . . . 505
8.548.1 Detailed Description . . . 506
8.548.2 Field Documentation . . . 506
8.548.2.1 Tlvresult . . . 506
8.549 unpack_nas_GetCDMANetworkParameters_t Struct Reference . . . 506
8.549.1 Detailed Description . . . 506
8.549.2 Field Documentation . . . 508
8.549.2.1 Application . . . 508
8.549.2.2 Broadcast . . . 508
8.549.2.3 CustomSCP . . . 508
8.549.2.4 ForceRev0 . . . 508
8.549.2.5 Protocol . . . 508
8.549.2.6 RegForeignNID . . . 508
8.549.2.7 RegForeignSID . . . 508
8.549.2.8 RegHomeSID . . . 508

8.549.2.9 Roaming . . . 508
8.549.2.10SCI . . . 508
8.549.2.11SCM . . . 508
8.550unpack_nas_GetHomeNetwork3GPP2_t Struct Reference . . . 508
8.550.1 Detailed Description . . . 509
8.550.2 Field Documentation . . . 510
8.550.2.1 nameSize . . . 510
8.550.2.2 pMCC . . . 510
8.550.2.3 pMNC . . . 510
8.550.2.4 pName . . . 510
8.550.2.5 pNID . . . 510
8.550.2.6 pNw2DescDisp . . . 510
8.550.2.7 pNw2DescEnc . . . 510
8.550.2.8 pNw2DescLen . . . 510
8.550.2.9 pNw2MCC . . . 510
8.550.2.10pNw2MNC . . . 510
8.550.2.11pNw2Name . . . 510
8.550.2.12pSID . . . 510
8.551unpack_nas_GetHomeNetwork_t Struct Reference . . . 510
8.551.1 Detailed Description . . . 510
8.551.2 Field Documentation . . . 511
8.551.2.1 mcc . . . 511
8.551.2.2 mnc . . . 511
8.551.2.3 name . . . 511
8.551.2.4 nid . . . 511
8.551.2.5 sid . . . 511
8.552unpack_nas_GetNetworkPreference_t Struct Reference . . . 511
8.552.1 Detailed Description . . . 511
8.552.2 Field Documentation . . . 512
8.552.2.1 ActiveTechPref . . . 512
8.552.2.2 Duration . . . 512
8.552.2.3 PersistentTechPref . . . 512
8.552.2.4 Tlvresult . . . 512
8.553unpack_nas_GetRFInfo_t Struct Reference . . . 512
8.553.1 Detailed Description . . . 512
8.553.2 Field Documentation . . . 513
8.553.2.1 instancesSize . . . 513
8.553.2.2 RFBandInfoElements . . . 513
8.554unpack_nas_GetServingNetwork_t Struct Reference . . . 513
8.554.1 Detailed Description . . . 513

8.554.2 Field Documentation . . . 515
8.554.2.1 CSDomain . . . 515
8.554.2.2 DataCaps . . . 515
8.554.2.3 DataCapsLen . . . 515
8.554.2.4 MCC . . . 515
8.554.2.5 MNC . . . 515
8.554.2.6 Name . . . 515
8.554.2.7 nameSize . . . 515
8.554.2.8 PSDomain . . . 515
8.554.2.9 RadioIfaces . . . 515
8.554.2.10 RadioIfacesSize . . . 515
8.554.2.11 RAN . . . 515
8.554.2.12 RegistrationState . . . 515
8.554.2.13 Roaming . . . 515
8.555 unpack_nas_GetServingNetworkCapabilities_t Struct Reference . . . 515
8.555.1 Detailed Description . . . 516
8.555.2 Field Documentation . . . 516
8.555.2.1 DataCaps . . . 516
8.555.2.2 DataCapsLen . . . 516
8.556 unpack_nas_GetSignalStrengths_t Struct Reference . . . 516
8.556.1 Detailed Description . . . 516
8.556.2 Field Documentation . . . 516
8.556.2.1 len . . . 516
8.556.2.2 radio . . . 516
8.556.2.3 rssi . . . 517
8.557 unpack_nas_PerformNetworkScan_t Struct Reference . . . 517
8.557.1 Detailed Description . . . 517
8.557.2 Field Documentation . . . 517
8.557.2.1 p3GppNetworkInfoInstances . . . 518
8.557.2.2 p3GppNetworkInstanceSize . . . 518
8.557.2.3 pPCSInstance . . . 518
8.557.2.4 pPCSInstanceSize . . . 518
8.557.2.5 pRATINstance . . . 518
8.557.2.6 pRATInstanceSize . . . 518
8.557.2.7 pScanResult . . . 518
8.558 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference . . . 518
8.558.1 Detailed Description . . . 518
8.558.2 Field Documentation . . . 518
8.558.2.1 dataCaps . . . 518
8.558.2.2 dataCapsSize . . . 518

8.559 unpack_nas_SetEventReportInd_t Struct Reference . . . 518
8.559.1 Detailed Description . . . 518
8.559.2 Field Documentation . . . 519
8.559.2.1 RFTlv . . . 519
8.559.2.2 RRTlv . . . 519
8.559.2.3 SLQSSSTlv . . . 519
8.559.2.4 SSTlv . . . 519
8.560 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference . . . 519
8.560.1 Detailed Description . . . 519
8.560.2 Field Documentation . . . 519
8.560.2.1 sPhyCaAggPcellInfo . . . 519
8.560.2.2 sPhyCaAggScellDlBw . . . 519
8.560.2.3 sPhyCaAggScellIndex . . . 519
8.560.2.4 sPhyCaAggScellIndType . . . 519
8.560.2.5 sPhyCaAggScellInfo . . . 519
8.561 unpack_nas_SetNetworkPreference_t Struct Reference . . . 520
8.561.1 Detailed Description . . . 520
8.561.2 Field Documentation . . . 520
8.561.2.1 Tlvresult . . . 520
8.562 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference . . . 520
8.562.1 Detailed Description . . . 520
8.562.2 Field Documentation . . . 521
8.562.2.1 roaming . . . 521
8.563 unpack_nas_SetServingSystemCallback_ind_t Struct Reference . . . 521
8.563.1 Detailed Description . . . 521
8.563.2 Field Documentation . . . 521
8.563.2.1 SSInfo . . . 521
8.563.2.2 Tlvresult . . . 521
8.564 unpack_nas_SLQSGetErrorRate_t Struct Reference . . . 521
8.564.1 Detailed Description . . . 521
8.564.2 Field Documentation . . . 522
8.564.2.1 pCDMAFrameErrRate . . . 522
8.564.2.2 pGSMBER . . . 522
8.564.2.3 pHDRPackErrRate . . . 522
8.564.2.4 pWCDMABER . . . 522
8.565 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference . . . 522
8.565.1 Detailed Description . . . 522
8.565.2 Field Documentation . . . 523
8.565.2.1 LTECphyCaInfo . . . 523
8.565.2.2 Tlvresult . . . 523

8.566 unpack_nas_SLQSGetNetworkTime_t Struct Reference . . . 523
8.566.1 Detailed Description . . . 523
8.566.2 Field Documentation . . . 523
8.566.2.1 p3GPP2TimeInfo . . . 523
8.566.2.2 p3GPPTimeInfo . . . 523
8.567 unpack_nas_SLQSGetOperatorNameData_t Struct Reference . . . 523
8.567.1 Detailed Description . . . 523
8.567.2 Field Documentation . . . 524
8.567.2.1 pNITZInformation . . . 524
8.567.2.2 pOperatorNameString . . . 524
8.567.2.3 pOperatorPLMNList . . . 524
8.567.2.4 pPLMNNetworkName . . . 524
8.567.2.5 pSrvcProviderName . . . 524
8.568 unpack_nas_SLQSGetPLMNName_t Struct Reference . . . 524
8.568.1 Field Documentation . . . 524
8.568.1.1 longName . . . 524
8.568.1.2 longNameCI . . . 525
8.568.1.3 longNameEn . . . 525
8.568.1.4 longNameLen . . . 525
8.568.1.5 longNameSB . . . 525
8.568.1.6 shortName . . . 525
8.568.1.7 shortNameCI . . . 525
8.568.1.8 shortNameEn . . . 525
8.568.1.9 shortNameLen . . . 525
8.568.1.10 shortNameSB . . . 525
8.568.1.11 spn . . . 525
8.568.1.12 spnEncoding . . . 525
8.568.1.13 spnLength . . . 525
8.569 unpack_nas_SLQSGetServingSystem_t Struct Reference . . . 525
8.569.1 Detailed Description . . . 526
8.569.2 Field Documentation . . . 528
8.569.2.1 BasestationID . . . 528
8.569.2.2 BasestationLatitude . . . 528
8.569.2.3 BasestationLongitude . . . 528
8.569.2.4 CallBarStatus . . . 528
8.569.2.5 CDMA_P_Rev . . . 528
8.569.2.6 CDMASystemInfoExt . . . 528
8.569.2.7 CellID . . . 528
8.569.2.8 ConcSvcInfo . . . 528
8.569.2.9 CurrentPLMN . . . 528

8.569.2.10 DataSrvCapabilities . . . 528
8.569.2.11 DefaultRoamInd . . . 528
8.569.2.12 DetailedSvcInfo . . . 528
8.569.2.13 DTMInd . . . 528
8.569.2.14 Gpp2TimeZone . . . 528
8.569.2.15 GppNetworkDSTAdjustment . . . 528
8.569.2.16 GppTimeZone . . . 528
8.569.2.17 HdrPersonality . . . 528
8.569.2.18 Lac . . . 528
8.569.2.19 NetworkID . . . 529
8.569.2.20 PRLInd . . . 529
8.569.2.21 RoamIndicatorVal . . . 529
8.569.2.22 RoamingIndicatorList . . . 529
8.569.2.23 ServingSystem . . . 529
8.569.2.24 SystemID . . . 529
8.569.2.25 TrackAreaCode . . . 529
8.570 unpack_nas_SLQSGetSignalStrength_t Struct Reference . . . 529
8.570.1 Detailed Description . . . 529
8.570.2 Field Documentation . . . 531
8.570.2.1 ecioList . . . 531
8.570.2.2 ecioListLen . . . 531
8.570.2.3 errorRateList . . . 531
8.570.2.4 errorRateListLen . . . 531
8.570.2.5 Io . . . 531
8.570.2.6 ltersrp . . . 531
8.570.2.7 ltesnr . . . 531
8.570.2.8 rsrqInfo . . . 531
8.570.2.9 rxSignalStrengthList . . . 531
8.570.2.10 rxSignalStrengthListLen . . . 531
8.570.2.11 signalStrengthReqMask . . . 531
8.570.2.12 sinr . . . 531
8.571 unpack_nas_SLQSGetSysInfo_t Struct Reference . . . 531
8.571.1 Detailed Description . . . 532
8.571.2 Field Documentation . . . 533
8.571.2.1 pAddCDMASysInfo . . . 533
8.571.2.2 pAddGSMSysInfo . . . 533
8.571.2.3 pAddHDRSysInfo . . . 533
8.571.2.4 pAddLTESysInfo . . . 533
8.571.2.5 pAddWCDMASysInfo . . . 534
8.571.2.6 pCDMASrvStatusInfo . . . 534

8.571.2.7 pCDMASysInfo . . . 534
8.571.2.8 pGSMCallBarringSysInfo . . . 534
8.571.2.9 pGSMCipherDomainSysInfo . . . 534
8.571.2.10 pGSMSrvStatusInfo . . . 534
8.571.2.11 pGSMSysInfo . . . 534
8.571.2.12 pHDRSrvStatusInfo . . . 534
8.571.2.13 pHDRSysInfo . . . 534
8.571.2.14 pLTESrvStatusInfo . . . 534
8.571.2.15 pLTESysInfo . . . 534
8.571.2.16 pLTEVoiceSupportSysInfo . . . 534
8.571.2.17 pWCDMACallBarringSysInfo . . . 534
8.571.2.18 pWCDMACipherDomainSysInfo . . . 534
8.571.2.19 pWCDMASrvStatusInfo . . . 534
8.571.2.20 pWCDMASysInfo . . . 534
8.572 unpack_nas_SLQSGetSysSelectionPref_t Struct Reference . . . 534
8.572.1 Detailed Description . . . 534
8.572.2 Field Documentation . . . 538
8.572.2.1 pBandPref . . . 538
8.572.2.2 pEmerMode . . . 538
8.572.2.3 pGWAcqOrderPref . . . 538
8.572.2.4 pLTEBandPref . . . 538
8.572.2.5 pModePref . . . 538
8.572.2.6 pNetSelPref . . . 538
8.572.2.7 pPRLPref . . . 538
8.572.2.8 pRoamPref . . . 538
8.572.2.9 pSrvDomainPref . . . 538
8.573 unpack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 538
8.573.1 Detailed Description . . . 539
8.573.2 Field Documentation . . . 539
8.573.2.1 pCDMAChannel . . . 539
8.573.2.2 pDirNum . . . 539
8.573.2.3 pHomeSIDNID . . . 539
8.573.2.4 pMinBasedIMSI . . . 539
8.573.2.5 pNAMNameInfo . . . 539
8.573.2.6 pTrueIMSI . . . 539
8.574 unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference . . . 539
8.574.1 Detailed Description . . . 540
8.574.2 Field Documentation . . . 540
8.574.2.1 pCDMAInfo . . . 540
8.574.2.2 pGERANInfo . . . 540

8.574.2.3 pLTEInfoInterfreq . . . 540
8.574.2.4 pLTEInfoIntrafreq . . . 540
8.574.2.5 pLTEInfoNeighboringGSM . . . 540
8.574.2.6 pLTEInfoNeighboringWCDMA . . . 540
8.574.2.7 pUMTSCellID . . . 540
8.574.2.8 pUMTSInfo . . . 540
8.574.2.9 pWCDMAInfoLTENeighborCell . . . 540
8.575 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference . . . 541
8.575.1 Detailed Description . . . 541
8.575.2 Field Documentation . . . 541
8.575.2.1 pColorCode . . . 541
8.576 unpack_nas_SLQSNasGetSigInfo_t Struct Reference . . . 541
8.576.1 Detailed Description . . . 541
8.576.2 Field Documentation . . . 542
8.576.2.1 CDMASSInfo . . . 542
8.576.2.2 GSMSSInfo . . . 542
8.576.2.3 HDRSSInfo . . . 542
8.576.2.4 LTESSInfo . . . 542
8.576.2.5 WCDMASSInfo . . . 542
8.577 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 542
8.577.1 Detailed Description . . . 542
8.577.2 Field Documentation . . . 542
8.577.2.1 pRXChain0Info . . . 542
8.577.2.2 pRXChain1Info . . . 542
8.577.2.3 pTXInfo . . . 542
8.578 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference . . . 542
8.578.1 Detailed Description . . . 543
8.578.2 Field Documentation . . . 543
8.578.2.1 pDayltSavAdj . . . 543
8.578.2.2 pRadioInterface . . . 543
8.578.2.3 pTimeZone . . . 543
8.578.2.4 universalTime . . . 543
8.579 unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference . . . 543
8.579.1 Detailed Description . . . 544
8.579.2 Field Documentation . . . 544
8.579.2.1 pCDMASigInfo . . . 544
8.579.2.2 pGSMSigInfo . . . 544
8.579.2.3 pHDRSigInfo . . . 544
8.579.2.4 pLTESigInfo . . . 544
8.579.2.5 pRscp . . . 544

8.579.2.6 pTDSCDMASigInfoExt . . . 544
8.579.2.7 pWCDMASigInfo . . . 544
8.580 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference . . . 544
8.580.1 Detailed Description . . . 545
8.580.2 Field Documentation . . . 545
8.580.2.1 pLteEARFCN . . . 545
8.580.2.2 pLtePCI . . . 545
8.580.2.3 pWcdmaUARFCN . . . 545
8.581 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference . . . 545
8.581.1 Detailed Description . . . 545
8.581.2 Field Documentation . . . 545
8.581.2.1 commonInfo . . . 546
8.581.2.2 pLTEInfo . . . 546
8.582 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference . . . 546
8.582.1 Detailed Description . . . 546
8.582.2 Field Documentation . . . 546
8.582.2.1 Info . . . 546
8.582.2.2 Tlvresult . . . 546
8.583 unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference . . . 546
8.583.1 Detailed Description . . . 546
8.583.2 Field Documentation . . . 547
8.583.2.1 t3396_apn . . . 547
8.583.2.2 t3396_plmn_id . . . 547
8.583.2.3 t3396_val . . . 547
8.584 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference . . . 547
8.584.1 Detailed Description . . . 547
8.584.2 Field Documentation . . . 547
8.584.2.1 Info . . . 547
8.584.2.2 Tlvresult . . . 547
8.585 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference . . . 547
8.585.1 Detailed Description . . . 547
8.585.2 Field Documentation . . . 548
8.585.2.1 pCurrentPersonality . . . 548
8.585.2.2 pPersonalityListLength . . . 548
8.585.2.3 pProtocolSubtypeElement . . . 548
8.586 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference . . . 548
8.586.1 Detailed Description . . . 548
8.586.2 Field Documentation . . . 548
8.586.2.1 pAppSubType . . . 548
8.586.2.2 pCurrentPrsnlty . . . 548

8.586.2.3 pPersonalityListLength . . . 548
8.586.2.4 pProtoSubTypElmnt . . . 548
8.587 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference . . . 549
8.587.1 Detailed Description . . . 549
8.587.2 Field Documentation . . . 549
8.587.2.1 pDRCParams . . . 549
8.587.2.2 pPilotSetData . . . 549
8.587.2.3 pUATI . . . 549
8.588 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference . . . 549
8.588.1 Detailed Description . . . 549
8.588.2 Field Documentation . . . 550
8.588.2.1 CQIValueCW0 . . . 550
8.588.2.2 CQIValueCW1 . . . 550
8.588.2.3 ValidityCW0 . . . 550
8.588.2.4 ValidityCW1 . . . 550
8.589 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference . . . 550
8.589.1 Detailed Description . . . 550
8.589.2 Field Documentation . . . 550
8.589.2.1 pSccRxInfo . . . 550
8.590 unpack_nas_SLQSSwiNetworkDebug_t Struct Reference . . . 550
8.590.1 Detailed Description . . . 551
8.590.2 Field Documentation . . . 551
8.590.2.1 pDataStatusDetail . . . 551
8.590.2.2 pDeviceConfigDetail . . . 551
8.590.2.3 pNetworkStat1x . . . 551
8.590.2.4 pNetworkStatEVDO . . . 551
8.590.2.5 pObjectVer . . . 551
8.591 unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference . . . 551
8.591.1 Detailed Description . . . 552
8.591.2 Field Documentation . . . 553
8.591.2.1 pAddCDMASysInfo . . . 553
8.591.2.2 pAddGSMSysInfo . . . 553
8.591.2.3 pAddHDRSysInfo . . . 553
8.591.2.4 pAddLTESysInfo . . . 553
8.591.2.5 pAddWCDMASysInfo . . . 553
8.591.2.6 pCDMASrvStatusInfo . . . 553
8.591.2.7 pCDMASysInfo . . . 554
8.591.2.8 pGSMCallBarringSysInfo . . . 554
8.591.2.9 pGSMCipherDomainSysInfo . . . 554
8.591.2.10 pGSMSrvStatusInfo . . . 554

8.591.2.11 pGSMSysInfo . . . 554
8.591.2.12 pHDRSrvStatusInfo . . . 554
8.591.2.13 pHDRSysInfo . . . 554
8.591.2.14 pLTESrvStatusInfo . . . 554
8.591.2.15 pLTESysInfo . . . 554
8.591.2.16 pLTEVoiceSupportSysInfo . . . 554
8.591.2.17 pSysInfoNoChange . . . 554
8.591.2.18 pWCDMACallBarringSysInfo . . . 554
8.591.2.19 pWCDMACipherDomainSysInfo . . . 554
8.591.2.20 pWCDMASrvStatusInfo . . . 554
8.591.2.21 pWCDMASysInfo . . . 554
8.592 unpack_omaDmConfigTlv_t Struct Reference . . . 554
8.592.1 Detailed Description . . . 554
8.592.2 Field Documentation . . . 555
8.592.2.1 alertmsg . . . 555
8.592.2.2 alertmsglength . . . 555
8.592.2.3 state . . . 555
8.592.2.4 userInputReq . . . 555
8.592.2.5 userInputTimeout . . . 555
8.593 unpack_omaDmFotaTlv_t Struct Reference . . . 555
8.593.1 Detailed Description . . . 555
8.593.2 Field Documentation . . . 557
8.593.2.1 description . . . 557
8.593.2.2 descriptionlength . . . 557
8.593.2.3 fwdloadsize . . . 557
8.593.2.4 fwloadComplete . . . 557
8.593.2.5 namelength . . . 557
8.593.2.6 package_name . . . 557
8.593.2.7 sessionType . . . 557
8.593.2.8 severity . . . 557
8.593.2.9 state . . . 557
8.593.2.10 updateCompleteStatus . . . 557
8.593.2.11 userInputReq . . . 557
8.593.2.12 userInputTimeout . . . 557
8.593.2.13 version . . . 557
8.593.2.14 versionlength . . . 557
8.594 unpack_omaDmNotificationsTlv_t Struct Reference . . . 557
8.594.1 Detailed Description . . . 557
8.594.2 Field Documentation . . . 558
8.594.2.1 notification . . . 558

8.594.2.2 sessionStatus . . . 558
8.595 unpack_pds_GetPDSDefaults_t Struct Reference . . . 558
8.595.1 Detailed Description . . . 558
8.595.2 Field Documentation . . . 559
8.595.2.1 pAccuracy . . . 559
8.595.2.2 pInterval . . . 559
8.595.2.3 pOperation . . . 559
8.595.2.4 pTimeout . . . 559
8.596 unpack_pds_GetPDSState_t Struct Reference . . . 559
8.596.1 Detailed Description . . . 559
8.596.2 Field Documentation . . . 559
8.596.2.1 pEnabledStatus . . . 559
8.596.2.2 pTrackingStatus . . . 559
8.597 unpack_pds_GetPortAutomaticTracking_t Struct Reference . . . 560
8.597.1 Detailed Description . . . 560
8.597.2 Field Documentation . . . 560
8.597.2.1 pbAuto . . . 560
8.598 unpack_pds_GetServiceAutomaticTracking_t Struct Reference . . . 560
8.598.1 Detailed Description . . . 560
8.598.2 Field Documentation . . . 560
8.598.2.1 pbAuto . . . 560
8.599 unpack_pds_GetXTRAAutomaticDownload_t Struct Reference . . . 560
8.599.1 Detailed Description . . . 561
8.599.2 Field Documentation . . . 561
8.599.2.1 pbEnabled . . . 561
8.599.2.2 pInterval . . . 561
8.600 unpack_pds_GetXTRANetwork_t Struct Reference . . . 561
8.600.1 Detailed Description . . . 561
8.600.2 Field Documentation . . . 561
8.600.2.1 pPreference . . . 561
8.601 unpack_pds_GetXTRAValidity_t Struct Reference . . . 561
8.601.1 Detailed Description . . . 562
8.601.2 Field Documentation . . . 562
8.601.2.1 pDuration . . . 562
8.601.2.2 pGPSWeek . . . 562
8.601.2.3 pGPSWeekOffset . . . 562
8.602 unpack_pds_SetEventReport_Ind_t Struct Reference . . . 562
8.602.1 Detailed Description . . . 562
8.602.2 Field Documentation . . . 563
8.602.2.1 dLatitude . . . 563

8.602.2.2 dLongitude . . . 563
8.602.2.3 has_dLatitude . . . 563
8.602.2.4 has_dLongitude . . . 564
8.602.2.5 has_PositionDataNMEA . . . 564
8.602.2.6 has_posSrc . . . 564
8.602.2.7 has_SessionStatus . . . 564
8.602.2.8 PositionDataNMEA . . . 564
8.602.2.9 posSrc . . . 564
8.602.2.10 SessionStatus . . . 564
8.603 unpack_pds_SetPdsState_Ind_t Struct Reference . . . 564
8.603.1 Detailed Description . . . 564
8.603.2 Field Documentation . . . 564
8.603.2.1 EnabledStatus . . . 564
8.603.2.2 TlvPresent . . . 564
8.603.2.3 TrackingStatus . . . 564
8.604 unpack_pds_SLQSGetAGPSConfig_t Struct Reference . . . 564
8.604.1 Detailed Description . . . 565
8.604.2 Field Documentation . . . 565
8.604.2.1 pServerAddress . . . 565
8.604.2.2 pServerPort . . . 565
8.604.2.3 pServerURL . . . 565
8.604.2.4 pServerURLLength . . . 565
8.605 unpack_pds_SLQSGetGPSStateInfo_t Struct Reference . . . 565
8.605.1 Detailed Description . . . 566
8.605.2 Field Documentation . . . 568
8.605.2.1 Altitude . . . 568
8.605.2.2 EngineState . . . 568
8.605.2.3 glo_almanac_sv_msk . . . 568
8.605.2.4 glo_ephemeris_sv_msk . . . 568
8.605.2.5 glo_health_sv_msk . . . 568
8.605.2.6 glo_visible_sv_msk . . . 568
8.605.2.7 gps_almanac_sv_msk . . . 568
8.605.2.8 gps_ephemeris_sv_msk . . . 569
8.605.2.9 gps_health_sv_msk . . . 569
8.605.2.10 gps_visible_sv_msk . . . 569
8.605.2.11 HorizontalUncertainty . . . 569
8.605.2.12 iono_valid . . . 569
8.605.2.13 Latitude . . . 569
8.605.2.14 Longitude . . . 569
8.605.2.15 sbas_almanac_sv_msk . . . 569

8.605.2.16 sbas_ephemeris_sv_msk . . . 569
8.605.2.17 sbas_health_sv_msk . . . 569
8.605.2.18 sbas_visible_sv_msk . . . 569
8.605.2.19 Time_uncert_ms . . . 569
8.605.2.20 TimeStmp_gps_week . . . 569
8.605.2.21 TimeStmp_tow_ms . . . 569
8.605.2.22 ValidMask . . . 569
8.605.2.23 VerticalUncertainty . . . 569
8.605.2.24 xtra_start_gps_minutes . . . 569
8.605.2.25 xtra_start_gps_week . . . 569
8.605.2.26 xtra_valid_duration_hours . . . 569
8.606 unpack_qmi_t Struct Reference . . . 569
8.606.1 Detailed Description . . . 569
8.606.2 Field Documentation . . . 570
8.606.2.1 msgid . . . 570
8.606.2.2 type . . . 570
8.606.2.3 xid . . . 570
8.607 unpack_qos_dataRate_t Struct Reference . . . 570
8.607.1 Detailed Description . . . 570
8.607.2 Field Documentation . . . 570
8.607.2.1 dataRateMax . . . 570
8.607.2.2 guaranteedRate . . . 570
8.608 unpack_qos_IPv4Addr_t Struct Reference . . . 570
8.608.1 Detailed Description . . . 570
8.608.2 Field Documentation . . . 570
8.608.2.1 addr . . . 571
8.608.2.2 subnetMask . . . 571
8.609 unpack_qos_IPv6Addr_t Struct Reference . . . 571
8.609.1 Detailed Description . . . 571
8.609.2 Field Documentation . . . 571
8.609.2.1 addr . . . 571
8.609.2.2 prefixLen . . . 571
8.610 unpack_qos_IPv6TrafCls_t Struct Reference . . . 571
8.610.1 Detailed Description . . . 571
8.610.2 Field Documentation . . . 571
8.610.2.1 mask . . . 572
8.610.2.2 val . . . 572
8.611 unpack_qos_pktErrRate_t Struct Reference . . . 572
8.611.1 Detailed Description . . . 572
8.611.2 Field Documentation . . . 572

8.611.2.1 exponent . . . 572
8.611.2.2 multiplier . . . 572
8.612 unpack_qos_Port_t Struct Reference . . . 572
8.612.1 Detailed Description . . . 572
8.612.2 Field Documentation . . . 572
8.612.2.1 port . . . 572
8.612.2.2 range . . . 572
8.613 unpack_qos_QosFlowInfo_t Struct Reference . . . 572
8.613.1 Detailed Description . . . 573
8.613.2 Field Documentation . . . 574
8.613.2.1 BearerID . . . 574
8.613.2.2 is_RxQFlowGranted_Available . . . 574
8.613.2.3 is_TxQFlowGranted_Available . . . 574
8.613.2.4 NumRxFilters . . . 574
8.613.2.5 NumTxFilters . . . 574
8.613.2.6 QFlowState . . . 574
8.613.2.7 RxQFilter . . . 574
8.613.2.8 RxQFlowGranted . . . 574
8.613.2.9 TxQFilter . . . 574
8.613.2.10 TxQFlowGranted . . . 574
8.614 unpack_qos_QosFlowInfoState_t Struct Reference . . . 574
8.614.1 Detailed Description . . . 574
8.614.2 Field Documentation . . . 574
8.614.2.1 id . . . 574
8.614.2.2 isNewFlow . . . 575
8.614.2.3 state . . . 575
8.615 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference . . . 575
8.615.1 Detailed Description . . . 575
8.615.2 Field Documentation . . . 575
8.615.2.1 NWQoSStatus . . . 575
8.616 unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 575
8.616.1 Detailed Description . . . 575
8.616.2 Field Documentation . . . 576
8.616.2.1 ambr_dl . . . 576
8.616.2.2 ambr_dl_ext . . . 576
8.616.2.3 ambr_dl_ext2 . . . 576
8.616.2.4 ambr_ul . . . 576
8.616.2.5 ambr_ul_ext . . . 576
8.616.2.6 ambr_ul_ext2 . . . 576
8.616.2.7 apnId . . . 576

8.617 unpack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 576
8.617.1 Detailed Description . . . 577
8.617.2 Field Documentation . . . 577
8.617.2.1 apnId . . . 577
8.617.2.2 numQosFlow . . . 577
8.617.2.3 qosFlow . . . 577
8.617.2.4 total_rx_bytes . . . 577
8.617.2.5 total_rx_pkt . . . 577
8.617.2.6 total_tx_bytes . . . 577
8.617.2.7 total_tx_bytes_drp . . . 577
8.617.2.8 total_tx_pkt . . . 577
8.617.2.9 total_tx_pkt_drp . . . 578
8.618 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference . . . 578
8.618.1 Detailed Description . . . 578
8.618.2 Field Documentation . . . 578
8.618.2.1 NumFlows . . . 578
8.618.2.2 QosFlowInfo . . . 578
8.619 unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference . . . 578
8.619.1 Detailed Description . . . 578
8.619.2 Field Documentation . . . 578
8.619.2.1 status . . . 579
8.620 unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference . . . 579
8.620.1 Detailed Description . . . 579
8.620.2 Field Documentation . . . 579
8.620.2.1 event . . . 579
8.621 unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference . . . 579
8.621.1 Detailed Description . . . 579
8.621.2 Field Documentation . . . 580
8.621.2.1 event . . . 580
8.621.2.2 id . . . 580
8.621.2.3 reason . . . 580
8.621.2.4 status . . . 580
8.622 unpack_qos_swiQosFilter_t Struct Reference . . . 580
8.622.1 Detailed Description . . . 581
8.622.2 Field Documentation . . . 582
8.622.2.1 EspSpi . . . 582
8.622.2.2 Id . . . 583
8.622.2.3 index . . . 583
8.622.2.4 IPv4DstAddr . . . 583
8.622.2.5 IPv4SrcAddr . . . 583

8.622.2.6 IPv4Tos . . . 583
8.622.2.7 IPv6DstAddr . . . 583
8.622.2.8 IPv6Label . . . 583
8.622.2.9 IPv6SrcAddr . . . 583
8.622.2.10 IPv6TrafCls . . . 583
8.622.2.11 is_EspSpi_Available . . . 583
8.622.2.12 is_Id_Available . . . 583
8.622.2.13 is_IPv4DstAddr_Available . . . 583
8.622.2.14 is_IPv4SrcAddr_Available . . . 583
8.622.2.15 is_IPv4Tos_Available . . . 583
8.622.2.16 is_IPv6DstAddr_Available . . . 583
8.622.2.17 is_IPv6Label_Available . . . 583
8.622.2.18 is_IPv6SrcAddr_Available . . . 583
8.622.2.19 is_IPv6TrafCls_Available . . . 583
8.622.2.20 is_NxtHdrProto_Available . . . 583
8.622.2.21 is_Precedence_Available . . . 583
8.622.2.22 is_TCPDstPort_Available . . . 583
8.622.2.23 is_TCPSrcPort_Available . . . 583
8.622.2.24 is_TranDstPort_Available . . . 583
8.622.2.25 is_TranSrcPort_Available . . . 583
8.622.2.26 is_UDPDstPort_Available . . . 583
8.622.2.27 is_UDPSrcPort_Available . . . 583
8.622.2.28 NxtHdrProto . . . 583
8.622.2.29 Precedence . . . 583
8.622.2.30 TCPDstPort . . . 584
8.622.2.31 TCPSrcPort . . . 584
8.622.2.32 TranDstPort . . . 584
8.622.2.33 TranSrcPort . . . 584
8.622.2.34 UDPDstPort . . . 584
8.622.2.35 UDPSrcPort . . . 584
8.622.2.36 version . . . 584
8.623 unpack_qos_swiQosFlow_t Struct Reference . . . 584
8.623.1 Detailed Description . . . 585
8.623.2 Field Documentation . . . 587
8.623.2.1 DataRate . . . 587
8.623.2.2 index . . . 587
8.623.2.3 is_DataRate_Available . . . 587
8.623.2.4 is_Jitter_Available . . . 587
8.623.2.5 is_Latency_Available . . . 587
8.623.2.6 is_LteQci_Available . . . 587

8.623.2.7 is_MaxAllowedPktSz_Available . . . 587
8.623.2.8 is_MinPolicedPktSz_Available . . . 587
8.623.2.9 is_PktErrRate_Available . . . 587
8.623.2.10 is_ProfileId3GPP2_Available . . . 587
8.623.2.11 is_TokenBucket_Available . . . 587
8.623.2.12 is_TrafficClass_Available . . . 587
8.623.2.13 is_val_3GPP2Pri_Available . . . 587
8.623.2.14 is_val_3GPPImCn_Available . . . 587
8.623.2.15 is_val_3GPPResResidualBER_Available . . . 587
8.623.2.16 is_val_3GPPSigInd_Available . . . 587
8.623.2.17 is_val_3GPPTraHdlPri_Available . . . 587
8.623.2.18 Jitter . . . 587
8.623.2.19 Latency . . . 587
8.623.2.20 LteQci . . . 587
8.623.2.21 MaxAllowedPktSz . . . 587
8.623.2.22 MinPolicedPktSz . . . 588
8.623.2.23 PktErrRate . . . 588
8.623.2.24 ProfileId3GPP2 . . . 588
8.623.2.25 TokenBucket . . . 588
8.623.2.26 TrafficClass . . . 588
8.623.2.27 val_3GPP2Pri . . . 588
8.623.2.28 val_3GPPImCn . . . 588
8.623.2.29 val_3GPPResResidualBER . . . 588
8.623.2.30 val_3GPPSigInd . . . 588
8.623.2.31 val_3GPPTraHdlPri . . . 588
8.624 unpack_qos_tokenBucket_t Struct Reference . . . 588
8.624.1 Detailed Description . . . 588
8.624.2 Field Documentation . . . 588
8.624.2.1 bucketSz . . . 588
8.624.2.2 peakRate . . . 588
8.624.2.3 tokenRate . . . 588
8.625 unpack_qos_Tos_t Struct Reference . . . 588
8.625.1 Detailed Description . . . 589
8.625.2 Field Documentation . . . 589
8.625.2.1 mask . . . 589
8.625.2.2 val . . . 589
8.626 unpack_QosFlowStat_t Struct Reference . . . 589
8.626.1 Detailed Description . . . 589
8.626.2 Field Documentation . . . 590
8.626.2.1 bearerId . . . 590

8.626.2.2 tx_bytes . . . 590
8.626.2.3 tx_bytes_drp . . . 590
8.626.2.4 tx_pkt . . . 590
8.626.2.5 tx_pkt_drp . . . 590
8.627 unpack_rms_GetSMSWake_t Struct Reference . . . 590
8.627.1 Detailed Description . . . 590
8.627.2 Field Documentation . . . 590
8.627.2.1 enabled . . . 590
8.627.2.2 wake_mask . . . 590
8.628 unpack_RMTransferStatistics_ind_t Struct Reference . . . 590
8.628.1 Detailed Description . . . 590
8.628.2 Field Documentation . . . 591
8.628.2.1 RxDropConutTlv . . . 591
8.628.2.2 RxOkByteCountTlv . . . 591
8.628.2.3 RxOkConutTlv . . . 591
8.628.2.4 TxDropConutTlv . . . 591
8.628.2.5 TxOkByteCountTlv . . . 591
8.628.2.6 TxOkConutTlv . . . 591
8.629 unpack_sar_SLQSGetRfSarState_t Struct Reference . . . 591
8.629.1 Detailed Description . . . 591
8.629.2 Field Documentation . . . 592
8.629.2.1 pSarRFState . . . 592
8.630 unpack_sms_GetSMSCAddress_t Struct Reference . . . 592
8.630.1 Detailed Description . . . 592
8.630.2 Field Documentation . . . 592
8.630.2.1 addressSize . . . 592
8.630.2.2 pSMSCAddress . . . 593
8.630.2.3 pSMSCType . . . 593
8.630.2.4 typeSize . . . 593
8.631 unpack_sms_SaveSMS_t Struct Reference . . . 593
8.631.1 Detailed Description . . . 593
8.631.2 Field Documentation . . . 593
8.631.2.1 pMessageIndex . . . 593
8.632 unpack_sms_SendSMS_t Struct Reference . . . 593
8.632.1 Detailed Description . . . 593
8.632.2 Field Documentation . . . 593
8.632.2.1 messageFailureCode . . . 593
8.632.2.2 messageID . . . 593
8.633 unpack_sms_SetNewSMSCallback_ind_t Struct Reference . . . 593
8.633.1 Detailed Description . . . 594

8.633.2 Field Documentation . . . 594
8.633.2.1 ETWSPLMNTlv . . . 594
8.633.2.2 ETWSTlv . . . 594
8.633.2.3 IMSTlv . . . 594
8.633.2.4 MMTlv . . . 594
8.633.2.5 NewMMTlv . . . 594
8.633.2.6 SMSCTlv . . . 595
8.633.2.7 TRMessageTlv . . . 595
8.634 unpack_sms_SetNewSMSCallback_t Struct Reference . . . 595
8.635 unpack_sms_SLQSDeleteSMS_t Struct Reference . . . 595
8.636 unpack_sms_SLQSGetIndicationRegister_t Struct Reference . . . 595
8.636.1 Detailed Description . . . 595
8.636.2 Field Documentation . . . 595
8.636.2.1 pGetIndicationRegInfo . . . 595
8.637 unpack_sms_SLQSGetMessageWaiting_t Struct Reference . . . 595
8.637.1 Detailed Description . . . 595
8.637.2 Field Documentation . . . 595
8.637.2.1 pGetMsgWaitingInfoResp . . . 595
8.638 unpack_sms_SLQSGetSMS_t Struct Reference . . . 596
8.638.1 Detailed Description . . . 596
8.638.2 Field Documentation . . . 596
8.638.2.1 message . . . 596
8.638.2.2 messageFormat . . . 596
8.638.2.3 messageSize . . . 596
8.638.2.4 messageTag . . . 596
8.639 unpack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 596
8.639.1 Detailed Description . . . 597
8.639.2 Field Documentation . . . 597
8.639.2.1 pBroadcastConfig . . . 597
8.639.2.2 pCDMABroadcastConfig . . . 597
8.640 unpack_sms_SLQSGetSMSList_t Struct Reference . . . 597
8.640.1 Detailed Description . . . 597
8.640.2 Field Documentation . . . 597
8.640.2.1 messageList . . . 597
8.640.2.2 messageListSize . . . 597
8.641 unpack_sms_SLQSGetTransLayerInfo_t Struct Reference . . . 597
8.641.1 Detailed Description . . . 598
8.641.2 Field Documentation . . . 598
8.641.2.1 pGetTransLayerInfo . . . 598
8.642 unpack_sms_SLQSGetTransNWRegInfo_t Struct Reference . . . 598

8.642.1 Detailed Description . . . 598
8.642.2 Field Documentation . . . 598
8.642.2.1 pGetTransNWRegInfo . . . 598
8.643 unpack_sms_SLQSModifySMSStatus_t Struct Reference . . . 598
8.644 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 598
8.644.1 Detailed Description . . . 598
8.644.2 Field Documentation . . . 599
8.644.2.1 pMaxStorageSizeResp . . . 599
8.645 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference . . . 599
8.645.1 Detailed Description . . . 599
8.645.2 Field Documentation . . . 599
8.645.2.1 pMessageProtocol . . . 599
8.646 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference . . . 599
8.646.1 Detailed Description . . . 599
8.646.2 Field Documentation . . . 600
8.646.2.1 pSmsStorage . . . 600
8.647 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference . . . 600
8.647.1 Detailed Description . . . 600
8.647.2 Field Documentation . . . 600
8.647.2.1 messageMode . . . 600
8.647.2.2 storageType . . . 600
8.648 unpack_swiavms_SLQSAVMSGetSettings_t Struct Reference . . . 600
8.648.1 Detailed Description . . . 600
8.648.2 Field Documentation . . . 601
8.648.2.1 AutoConnect . . . 601
8.648.2.2 AutoReboot . . . 601
8.648.2.3 FwAutodownload . . . 601
8.648.2.4 FwAutoSDM . . . 601
8.648.2.5 FwAutoUpdate . . . 601
8.648.2.6 OMADMEnabled . . . 601
8.648.2.7 pAPNInfo . . . 601
8.648.2.8 pConnectionRetryTimers . . . 601
8.648.2.9 pNotificationStore . . . 601
8.648.2.10 pPeroidsInfo . . . 601
8.648.2.11 pPollingTimer . . . 601
8.648.2.12 resultcode . . . 601
8.649 unpack_swiavms_SLQSAVMSSendSelection_t Struct Reference . . . 601
8.649.1 Detailed Description . . . 601
8.649.2 Field Documentation . . . 601
8.649.2.1 resultcode . . . 601

8.650 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference . . . 601
8.650.1 Detailed Description . . . 602
8.650.2 Field Documentation . . . 602
8.650.2.1 resultcode . . . 602
8.651 unpack_swiavms_SLQSAVMSSetSettings_t Struct Reference . . . 602
8.651.1 Detailed Description . . . 602
8.651.2 Field Documentation . . . 602
8.651.2.1 resultcode . . . 602
8.652 unpack_swiavms_SLQSAVMSStartSession_t Struct Reference . . . 602
8.652.1 Detailed Description . . . 602
8.652.2 Field Documentation . . . 602
8.652.2.1 resultcode . . . 603
8.652.2.2 sessionResponse . . . 603
8.653 unpack_swiavms_SLQSAVMSStopSession_t Struct Reference . . . 603
8.653.1 Detailed Description . . . 603
8.653.2 Field Documentation . . . 603
8.653.2.1 resultcode . . . 603
8.654 unpack_swiloc_SwiLocGetAutoStart_t Struct Reference . . . 603
8.654.1 Detailed Description . . . 603
8.654.2 Field Documentation . . . 604
8.654.2.1 fix_rate . . . 604
8.654.2.2 fix_rate_reported . . . 604
8.654.2.3 fix_type . . . 604
8.654.2.4 fix_type_reported . . . 604
8.654.2.5 function . . . 604
8.654.2.6 function_reported . . . 604
8.654.2.7 max_dist . . . 604
8.654.2.8 max_dist_reported . . . 605
8.654.2.9 max_time . . . 605
8.654.2.10 max_time_reported . . . 605
8.655 unpack_swioma_SLQSOMADMAlertCallback_ind_t Struct Reference . . . 605
8.655.1 Detailed Description . . . 605
8.655.2 Field Documentation . . . 605
8.655.2.1 eventType . . . 605
8.655.2.2 SessionInfoConfig . . . 605
8.655.2.3 SessionInfoFota . . . 605
8.655.2.4 SessionInfoNotification . . . 605
8.656 unpack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 605
8.656.1 Detailed Description . . . 606
8.656.2 Field Documentation . . . 607

8.656.2.1 Date . . . 607
8.656.2.2 DateLength . . . 607
8.656.2.3 PkgDescLength . . . 607
8.656.2.4 PkgDescription . . . 608
8.656.2.5 PkgName . . . 608
8.656.2.6 PkgNameLength . . . 608
8.656.2.7 RetryCount . . . 608
8.656.2.8 SessionState . . . 608
8.656.2.9 SessionType . . . 608
8.656.2.10 Severity . . . 608
8.656.2.11 Source . . . 608
8.656.2.12 SourceLength . . . 608
8.656.2.13 Status . . . 608
8.656.2.14 Time . . . 608
8.656.2.15 TimeLength . . . 608
8.656.2.16 UpdateCompleteStatus . . . 608
8.657 unpack_swioma_SLQSOMADMGetSettings_t Struct Reference . . . 608
8.657.1 Detailed Description . . . 608
8.657.2 Field Documentation . . . 609
8.657.2.1 Autosdm . . . 609
8.657.2.2 FOTAdownload . . . 609
8.657.2.3 FOTAUpdate . . . 609
8.657.2.4 FwAutoCheck . . . 609
8.657.2.5 OMADMEnabled . . . 609
8.658 unpack_swioma_SLQSOMADMStartSession_t Struct Reference . . . 610
8.658.1 Detailed Description . . . 610
8.658.2 Field Documentation . . . 610
8.658.2.1 FwAvailability . . . 610
8.659 unpack_uim_ChangePin_t Struct Reference . . . 610
8.659.1 Detailed Description . . . 610
8.659.2 Field Documentation . . . 611
8.659.2.1 pEncryptedPIN1 . . . 611
8.659.2.2 pIndicationToken . . . 611
8.659.2.3 pRemainingRetries . . . 611
8.659.2.4 Tlvresult . . . 611
8.660 unpack_uim_GetCardStatus_t Struct Reference . . . 611
8.660.1 Detailed Description . . . 611
8.660.2 Field Documentation . . . 611
8.660.2.1 pCardStatus . . . 611
8.660.2.2 pHotSwapStatus . . . 611

8.660.2.3 TlvResult . . . 611
8.661 unpack_uim_ReadTransparent_t Struct Reference . . . 611
8.661.1 Detailed Description . . . 612
8.661.2 Field Documentation . . . 612
8.661.2.1 pCardResult . . . 612
8.661.2.2 pEncryptedData . . . 612
8.661.2.3 pIndicationToken . . . 612
8.661.2.4 pReadResult . . . 612
8.661.2.5 TlvResult . . . 612
8.662 unpack_uim_SetPinProtection_t Struct Reference . . . 612
8.662.1 Detailed Description . . . 612
8.662.2 Field Documentation . . . 613
8.662.2.1 pEncryptedPIN1 . . . 613
8.662.2.2 pIndicationToken . . . 613
8.662.2.3 pRemainingRetries . . . 613
8.662.2.4 TlvResult . . . 613
8.663 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference . . . 613
8.663.1 Detailed Description . . . 613
8.663.2 Field Documentation . . . 613
8.663.2.1 bNumberOfPhySlots . . . 613
8.663.2.2 slotsstatusChange . . . 613
8.664 unpack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 614
8.664.1 Detailed Description . . . 614
8.664.2 Field Documentation . . . 614
8.664.2.1 pAuthenticateResult . . . 614
8.664.2.2 pCardResult . . . 614
8.664.2.3 pIndicationToken . . . 614
8.665 unpack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 614
8.665.1 Detailed Description . . . 614
8.665.2 Field Documentation . . . 615
8.665.2.1 pRemainingRetries . . . 615
8.666 unpack_uim_SLQSUIMEventRegister_t Struct Reference . . . 615
8.666.1 Detailed Description . . . 615
8.666.2 Field Documentation . . . 615
8.666.2.1 eventMask . . . 615
8.667 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 615
8.667.1 Detailed Description . . . 615
8.667.2 Field Documentation . . . 616
8.667.2.1 pAutoSelection . . . 616
8.667.2.2 pHaltSubscription . . . 616

8.667.2.3 pPersonalizationStatus . . . 616
8.668 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 616
8.668.1 Detailed Description . . . 616
8.668.2 Field Documentation . . . 616
8.668.2.1 pCardResult . . . 616
8.668.2.2 pFileAttributes . . . 616
8.668.2.3 pIndicationToken . . . 616
8.669 unpack_uim_SLQSUIMGetSlotsStatus_t Struct Reference . . . 617
8.669.1 Detailed Description . . . 617
8.669.2 Field Documentation . . . 617
8.669.2.1 pNumberOfPhySlot . . . 617
8.669.2.2 pUimSlotsStatus . . . 617
8.670 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 617
8.670.1 Detailed Description . . . 617
8.670.2 Field Documentation . . . 617
8.670.2.1 pRefreshEvent . . . 617
8.671 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference . . . 617
8.671.1 Detailed Description . . . 618
8.671.2 Field Documentation . . . 618
8.671.2.1 pCardStatus . . . 618
8.672 unpack_uim_UnblockPin_t Struct Reference . . . 618
8.672.1 Detailed Description . . . 618
8.672.2 Field Documentation . . . 618
8.672.2.1 pEncryptedPIN1 . . . 618
8.672.2.2 pIndicationToken . . . 618
8.672.2.3 pRemainingRetries . . . 618
8.672.2.4 Tlvresult . . . 618
8.673 unpack_uim_VerifyPin_t Struct Reference . . . 618
8.673.1 Detailed Description . . . 619
8.673.2 Field Documentation . . . 619
8.673.2.1 pEncryptedPIN1 . . . 619
8.673.2.2 pIndicationToken . . . 619
8.673.2.3 pRemainingRetries . . . 619
8.673.2.4 Tlvresult . . . 619
8.674 unpack_voice_SLQSOriginateUSSD_t Struct Reference . . . 619
8.674.1 Field Documentation . . . 619
8.674.1.1 pAlphaIDInfo . . . 619
8.674.1.2 pCallId . . . 620
8.674.1.3 pCcResultType . . . 620
8.674.1.4 pCCSuppsType . . . 620

8.674.1.5 pfailureCause . . . 620
8.674.1.6 pUSSDInfo . . . 620
8.675 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference . . . 620
8.675.1 Detailed Description . . . 620
8.675.2 Field Documentation . . . 620
8.675.2.1 pCallId . . . 620
8.676 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference . . . 620
8.676.1 Detailed Description . . . 620
8.676.2 Field Documentation . . . 620
8.676.2.1 pCallID . . . 621
8.677 unpack_voice_SLQSVoiceDialCall_t Struct Reference . . . 621
8.677.1 Detailed Description . . . 621
8.677.2 Field Documentation . . . 621
8.677.2.1 pAlphaIDInfo . . . 621
8.677.2.2 pCallID . . . 621
8.677.2.3 pCCResultType . . . 621
8.677.2.4 pCCSUPSType . . . 621
8.678 unpack_voice_SLQSVoiceEndCall_t Struct Reference . . . 621
8.678.1 Detailed Description . . . 622
8.678.2 Field Documentation . . . 622
8.678.2.1 pCallId . . . 622
8.679 unpack_voice_SLQSVoiceGetAllCallInfo_t Struct Reference . . . 622
8.679.1 Detailed Description . . . 622
8.679.2 Field Documentation . . . 624
8.679.2.1 pArrAlertingPattern . . . 624
8.679.2.2 pArrAlertingType . . . 624
8.679.2.3 pArrAlphaID . . . 624
8.679.2.4 pArrCalledPartyNum . . . 624
8.679.2.5 pArrCallEndReason . . . 624
8.679.2.6 pArrCallInfo . . . 624
8.679.2.7 pArrConnectPartyNum . . . 624
8.679.2.8 pArrDiagInfo . . . 624
8.679.2.9 pArrRedirPartyNum . . . 624
8.679.2.10 pArrRemotePartyName . . . 624
8.679.2.11 pArrRemotePartyNum . . . 624
8.679.2.12 pArrSvcOption . . . 624
8.679.2.13 pArrUUSInfo . . . 624
8.679.2.14 pOTASPStatus . . . 624
8.679.2.15 pVoicePrivacy . . . 624
8.680 unpack_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 624

8.680.1 Detailed Description . . . 625
8.680.2 Field Documentation . . . 625
8.680.2.1 pAlphaIDInfo . . . 625
8.680.2.2 pCallID . . . 625
8.680.2.3 pCCResType . . . 625
8.680.2.4 pCCSUPSType . . . 625
8.680.2.5 pFailCause . . . 625
8.680.2.6 pSvcClass . . . 626
8.681 unpack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 626
8.681.1 Detailed Description . . . 626
8.681.2 Field Documentation . . . 627
8.681.2.1 pAlphaIDInfo . . . 627
8.681.2.2 pCallID . . . 627
8.681.2.3 pCCResType . . . 627
8.681.2.4 pCCSUPSType . . . 627
8.681.2.5 pFailCause . . . 627
8.681.2.6 pGetCallFWExtInfo . . . 627
8.681.2.7 pGetCallFWInfo . . . 627
8.682 unpack_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 627
8.682.1 Detailed Description . . . 627
8.682.2 Field Documentation . . . 629
8.682.2.1 pAlertingPattern . . . 629
8.682.2.2 pAlertType . . . 629
8.682.2.3 pAlphaIDInfo . . . 629
8.682.2.4 pCallInfo . . . 629
8.682.2.5 pConnectNumInfo . . . 629
8.682.2.6 pDiagInfo . . . 629
8.682.2.7 pOTASPStatus . . . 629
8.682.2.8 pRemotePartyName . . . 629
8.682.2.9 pRemotePartyNum . . . 629
8.682.2.10 pSrvOpt . . . 629
8.682.2.11 pUUSInfo . . . 629
8.682.2.12 pVoicePrivacy . . . 629
8.683 unpack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 629
8.683.1 Detailed Description . . . 630
8.683.2 Field Documentation . . . 630
8.683.2.1 pAlphaIDInfo . . . 631
8.683.2.2 pCallID . . . 631
8.683.2.3 pCCResType . . . 631
8.683.2.4 pCCSUPSType . . . 631

8.683.2.5 pFailCause . . . 631
8.683.2.6 pSvcClass . . . 631
8.684 unpack_voice_SLQSVoiceGetCLIP_t Struct Reference . . . 631
8.684.1 Detailed Description . . . 631
8.684.2 Field Documentation . . . 632
8.684.2.1 pAlphaIDInfo . . . 632
8.684.2.2 pCallID . . . 632
8.684.2.3 pCCResType . . . 632
8.684.2.4 pCCSUPSType . . . 632
8.684.2.5 pCLIPResp . . . 632
8.684.2.6 pFailCause . . . 632
8.685 unpack_voice_SLQSVoiceGetCLIR_t Struct Reference . . . 632
8.685.1 Detailed Description . . . 632
8.685.2 Field Documentation . . . 633
8.685.2.1 pAlphaIDInfo . . . 633
8.685.2.2 pCallID . . . 633
8.685.2.3 pCCResType . . . 633
8.685.2.4 pCCSUPSType . . . 633
8.685.2.5 pCLIRResp . . . 633
8.685.2.6 pFailCause . . . 633
8.686 unpack_voice_SLQSVoiceGetCNAP_t Struct Reference . . . 633
8.686.1 Detailed Description . . . 633
8.686.2 Field Documentation . . . 634
8.686.2.1 pAlphaIDInfo . . . 634
8.686.2.2 pCallID . . . 634
8.686.2.3 pCCResType . . . 634
8.686.2.4 pCCSUPSType . . . 634
8.686.2.5 pCNAPResp . . . 634
8.686.2.6 pFailCause . . . 634
8.687 unpack_voice_SLQSVoiceGetCOLP_t Struct Reference . . . 634
8.687.1 Detailed Description . . . 635
8.687.2 Field Documentation . . . 635
8.687.2.1 pAlphaIDInfo . . . 635
8.687.2.2 pCallID . . . 636
8.687.2.3 pCCResType . . . 636
8.687.2.4 pCCSUPSType . . . 636
8.687.2.5 pCOLPResp . . . 636
8.687.2.6 pFailCause . . . 636
8.688 unpack_voice_SLQSVoiceGetCOLR_t Struct Reference . . . 636
8.688.1 Detailed Description . . . 636

8.688.2 Field Documentation . . . 637
8.688.2.1 pAlphaIDInfo . . . 637
8.688.2.2 pCallID . . . 637
8.688.2.3 pCCResType . . . 637
8.688.2.4 pCCSUPSType . . . 637
8.688.2.5 pCOLRResp . . . 637
8.688.2.6 pFailCause . . . 637
8.689 unpack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 637
8.689.1 Detailed Description . . . 637
8.689.2 Field Documentation . . . 638
8.689.2.1 pAirTimerCnt . . . 639
8.689.2.2 pAutoAnswerStat . . . 639
8.689.2.3 pCurAMRConfig . . . 639
8.689.2.4 pCurPrefVoiceSO . . . 639
8.689.2.5 pCurrTTYMode . . . 639
8.689.2.6 pCurVoiceDomainPref . . . 639
8.689.2.7 pCurVoicePrivacyPref . . . 639
8.689.2.8 pRoamTimerCnt . . . 639
8.690 unpack_voice_SLQSVoiceManageCalls_t Struct Reference . . . 639
8.690.1 Detailed Description . . . 639
8.690.2 Field Documentation . . . 639
8.690.2.1 pFailCause . . . 639
8.691 unpack_voice_SLQSVoiceSendFlash_t Struct Reference . . . 639
8.691.1 Detailed Description . . . 639
8.691.2 Field Documentation . . . 640
8.691.2.1 pCallID . . . 640
8.692 unpack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 640
8.692.1 Detailed Description . . . 640
8.692.2 Field Documentation . . . 640
8.692.2.1 pAlphaIDInfo . . . 641
8.692.2.2 pCallID . . . 641
8.692.2.3 pCCResType . . . 641
8.692.2.4 pCCSUPSType . . . 641
8.692.2.5 pFailCause . . . 641
8.693 unpack_voice_SLQSVoiceSetConfig_t Struct Reference . . . 641
8.693.1 Detailed Description . . . 641
8.693.2 Field Documentation . . . 642
8.693.2.1 pAirTimerStatus . . . 642
8.693.2.2 pAutoAnsStatus . . . 642
8.693.2.3 pPrefVoiceSOStatus . . . 642

8.693.2.4 pRoamTimerStatus . . . 642
8.693.2.5 pTTYConfigStatus . . . 642
8.693.2.6 pVoiceDomainPrefStatus . . . 642
8.694 unpack_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 642
8.694.1 Detailed Description . . . 643
8.694.2 Field Documentation . . . 643
8.694.2.1 pAlphaIDInfo . . . 643
8.694.2.2 pCallID . . . 643
8.694.2.3 pCCResultType . . . 643
8.694.2.4 pCCSUPSType . . . 643
8.694.2.5 pFailCause . . . 643
8.695 unpack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 643
8.695.1 Detailed Description . . . 644
8.695.2 Field Documentation . . . 644
8.695.2.1 pCallID . . . 644
8.696 unpack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 644
8.696.1 Detailed Description . . . 644
8.696.2 Field Documentation . . . 644
8.696.2.1 callID . . . 644
8.697 unpack_wds_DHCPv4ClientLease_ind_t Struct Reference . . . 644
8.697.1 Detailed Description . . . 645
8.697.2 Field Documentation . . . 645
8.697.2.1 DHCPv4LeaseOptTlv . . . 645
8.697.2.2 DHCPv4LeaseStateTlv . . . 645
8.697.2.3 IPv4AddrTlv . . . 645
8.697.2.4 ProfileIdTlv . . . 645
8.698 unpack_wds_GetAutoconnect_t Struct Reference . . . 645
8.698.1 Detailed Description . . . 645
8.698.2 Field Documentation . . . 645
8.698.2.1 psetting . . . 645
8.699 unpack_wds_GetByteTotals_t Struct Reference . . . 645
8.699.1 Detailed Description . . . 646
8.699.2 Field Documentation . . . 646
8.699.2.1 pRXTotalBytes . . . 646
8.699.2.2 pTXTotalBytes . . . 646
8.700 unpack_wds_GetConnectionRate_t Struct Reference . . . 646
8.700.1 Detailed Description . . . 646
8.700.2 Field Documentation . . . 646
8.700.2.1 currentChannelRXRate . . . 646
8.700.2.2 currentChannelTXRate . . . 646

8.700.2.3 maxChannelRXRate . . . 646
8.700.2.4 maxChannelTXRate . . . 646
8.701 unpack_wds_GetDataBearerTechnology_t Struct Reference . . . 647
8.701.1 Detailed Description . . . 647
8.701.2 Field Documentation . . . 647
8.701.2.1 pDataBearer . . . 647
8.702 unpack_wds_GetDefaultProfile_t Struct Reference . . . 647
8.702.1 Detailed Description . . . 648
8.702.2 Field Documentation . . . 649
8.702.2.1 apnname . . . 649
8.702.2.2 apnsize . . . 649
8.702.2.3 auth . . . 649
8.702.2.4 ipaddr . . . 649
8.702.2.5 ipaddrv6 . . . 649
8.702.2.6 name . . . 649
8.702.2.7 namesize . . . 649
8.702.2.8 pdptype . . . 649
8.702.2.9 pridns . . . 649
8.702.2.10 pridnsv6 . . . 649
8.702.2.11 secdns . . . 649
8.702.2.12 secdnsv6 . . . 649
8.702.2.13 username . . . 649
8.702.2.14 usersize . . . 649
8.703 unpack_wds_GetDefaultProfileNum_t Struct Reference . . . 649
8.703.1 Detailed Description . . . 650
8.703.2 Field Documentation . . . 650
8.703.2.1 index . . . 650
8.704 unpack_wds_GetDormancyState_t Struct Reference . . . 650
8.704.1 Detailed Description . . . 650
8.704.2 Field Documentation . . . 650
8.704.2.1 dormancyState . . . 650
8.705 unpack_wds_GetLastMobileIPError_t Struct Reference . . . 650
8.705.1 Detailed Description . . . 650
8.705.2 Field Documentation . . . 650
8.705.2.1 error . . . 650
8.706 unpack_wds_GetMobileIP_t Struct Reference . . . 651
8.706.1 Detailed Description . . . 651
8.706.2 Field Documentation . . . 651
8.706.2.1 mipMode . . . 651
8.707 unpack_wds_GetMobileIPProfile_t Struct Reference . . . 651

8.707.1 Detailed Description . . . 651
8.707.2 Field Documentation . . . 652
8.707.2.1 AAASPI . . . 652
8.707.2.2 AAAState . . . 652
8.707.2.3 address . . . 652
8.707.2.4 enabled . . . 652
8.707.2.5 HASPI . . . 653
8.707.2.6 HAState . . . 653
8.707.2.7 NAI . . . 653
8.707.2.8 naiSize . . . 653
8.707.2.9 primaryHA . . . 653
8.707.2.10 revTunneling . . . 653
8.707.2.11 secondaryHA . . . 653
8.708 unpack_wds_GetPacketStatistics_t Struct Reference . . . 653
8.708.1 Detailed Description . . . 653
8.708.2 Field Documentation . . . 654
8.708.2.1 pRXDroppedCount . . . 654
8.708.2.2 pRXOkBytesCount . . . 654
8.708.2.3 pRXOKBytesLastCall . . . 654
8.708.2.4 pRXPacketErrors . . . 654
8.708.2.5 pRXPacketOverflows . . . 654
8.708.2.6 pRXPacketSuccesses . . . 654
8.708.2.7 pTXDroppedCount . . . 654
8.708.2.8 pTXOkBytesCount . . . 654
8.708.2.9 pTXOKBytesLastCall . . . 654
8.708.2.10 pTXPacketErrors . . . 654
8.708.2.11 pTXPacketOverflows . . . 654
8.708.2.12 pTXPacketSuccesses . . . 654
8.709 unpack_wds_GetPacketStatus_t Struct Reference . . . 654
8.709.1 Detailed Description . . . 655
8.709.2 Field Documentation . . . 656
8.709.2.1 rXDroppedCount . . . 656
8.709.2.2 rXOkBytesCount . . . 656
8.709.2.3 rXOKBytesLastCall . . . 656
8.709.2.4 rXPacketErrors . . . 656
8.709.2.5 rXPacketOverflows . . . 656
8.709.2.6 rXPacketSuccesses . . . 656
8.709.2.7 tXDroppedCount . . . 656
8.709.2.8 tXOkBytesCount . . . 656
8.709.2.9 tXOKBytesLastCall . . . 656

8.709.2.10 TXPacketErrors . . . 656
8.709.2.11 TXPacketOverflows . . . 656
8.709.2.12 TXPacketSuccesses . . . 656
8.710 unpack_wds_GetSessionDuration_t Struct Reference . . . 656
8.710.1 Detailed Description . . . 656
8.710.2 Field Documentation . . . 656
8.710.2.1 callDuration . . . 656
8.711 unpack_wds_GetSessionState_t Struct Reference . . . 657
8.711.1 Detailed Description . . . 657
8.711.2 Field Documentation . . . 657
8.711.2.1 connectionStatus . . . 657
8.712 unpack_wds_RMSetTransferStatistics_t Struct Reference . . . 657
8.712.1 Detailed Description . . . 657
8.713 unpack_wds_SetMobileIPProfile_t Struct Reference . . . 657
8.713.1 Detailed Description . . . 657
8.714 unpack_wds_SLQSCreateProfile_t Struct Reference . . . 657
8.714.1 Detailed Description . . . 657
8.714.2 Field Documentation . . . 658
8.714.2.1 pCreateProfileOut . . . 658
8.714.2.2 pProfileID . . . 658
8.714.2.3 Tlvresult . . . 658
8.715 unpack_wds_SLQSDeleteProfile_t Struct Reference . . . 658
8.715.1 Detailed Description . . . 658
8.715.2 Field Documentation . . . 658
8.715.2.1 extendedErrorCode . . . 658
8.716 unpack_wds_SLQSGet3GPPConfigItem_t Struct Reference . . . 658
8.716.1 Detailed Description . . . 658
8.716.2 Field Documentation . . . 659
8.716.2.1 _3gppRelease . . . 659
8.716.2.2 defaultPDNEnabled . . . 659
8.716.2.3 LTEAttachProfile . . . 660
8.716.2.4 LTEAttachProfileList . . . 660
8.716.2.5 LTEAttachProfileListLen . . . 660
8.716.2.6 profileList . . . 660
8.717 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 660
8.717.1 Detailed Description . . . 660
8.717.2 Field Documentation . . . 660
8.717.2.1 currNetworkInfo . . . 660
8.717.2.2 networkInfoLen . . . 660
8.717.2.3 prefNetwork . . . 660

8.718 unpack_wds_SLQSGetCurrentChannelRate_t Struct Reference . . . 660
8.718.1 Detailed Description . . . 661
8.718.2 Field Documentation . . . 661
8.718.2.1 current_channel_rx_rate . . . 661
8.718.2.2 current_channel_tx_rate . . . 661
8.718.2.3 max_channel_rx_rate . . . 661
8.718.2.4 max_channel_tx_rate . . . 661
8.719 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 661
8.719.1 Detailed Description . . . 661
8.719.2 Field Documentation . . . 662
8.719.2.1 curDataBearerTechnology . . . 662
8.719.2.2 dataBearerMask . . . 662
8.719.2.3 lastCallDataBearerTechnology . . . 662
8.720 unpack_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 662
8.720.1 Detailed Description . . . 662
8.720.2 Field Documentation . . . 664
8.720.2.1 callEndReason . . . 664
8.720.2.2 channelRate . . . 664
8.720.2.3 connectionStatus . . . 664
8.720.2.4 dataBearerTech . . . 664
8.720.2.5 dormancyStatus . . . 664
8.720.2.6 lastCallDataBearerTech . . . 664
8.720.2.7 lastCallRXOKBytesCnt . . . 664
8.720.2.8 lastCallTXOKBytesCnt . . . 664
8.720.2.9 mdmCallDurationActive . . . 664
8.720.2.10 rxOKBytesCount . . . 665
8.720.2.11 txOKBytesCount . . . 665
8.721 unpack_wds_SLQSGetProfileSettings_t Struct Reference . . . 665
8.721.1 Detailed Description . . . 665
8.721.2 Field Documentation . . . 665
8.721.2.1 pProfileSettings . . . 665
8.721.2.2 ProfileType . . . 665
8.721.2.3 Tlvresult . . . 665
8.722 unpack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 665
8.722.1 Detailed Description . . . 666
8.722.2 Field Documentation . . . 666
8.722.2.1 APNName . . . 667
8.722.2.2 Authentication . . . 667
8.722.2.3 DomainList . . . 667
8.722.2.4 GPRSGrantedQoS . . . 667

8.722.2.5 GWAddressV4 . . . 667
8.722.2.6 IMCNflag . . . 667
8.722.2.7 IPFamilyPreference . . . 667
8.722.2.8 IPv4 . . . 667
8.722.2.9 IPV6AddrInfo . . . 667
8.722.2.10 IPV6GWAddrInfo . . . 667
8.722.2.11 Mtu . . . 667
8.722.2.12 PCSCFAddrPCO . . . 667
8.722.2.13 PCSCFFQDNAddrList . . . 667
8.722.2.14 PDPType . . . 667
8.722.2.15 PrimaryDNSV4 . . . 667
8.722.2.16 PrimaryDNSV6 . . . 667
8.722.2.17 ProfileID . . . 667
8.722.2.18 ProfileName . . . 667
8.722.2.19 SecondaryDNSV4 . . . 667
8.722.2.20 SecondaryDNSV6 . . . 667
8.722.2.21 ServerAddrList . . . 667
8.722.2.22 SubnetMaskV4 . . . 667
8.722.2.23 Technology . . . 667
8.722.2.24 UMTSGrantedQoS . . . 667
8.722.2.25 Username . . . 667
8.723 unpack_wds_SLQSModifyProfile_t Struct Reference . . . 667
8.723.1 Detailed Description . . . 668
8.723.2 Field Documentation . . . 668
8.723.2.1 pExtErrorCode . . . 668
8.724 unpack_wds_SLQSSetIPFamilyPreference_t Struct Reference . . . 668
8.724.1 Detailed Description . . . 668
8.724.2 Field Documentation . . . 668
8.724.2.1 Tlvresult . . . 668
8.725 unpack_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference . . . 668
8.725.1 Detailed Description . . . 668
8.725.2 Field Documentation . . . 669
8.725.2.1 bearerID . . . 669
8.725.2.2 conn_status . . . 669
8.725.2.3 ipFamily . . . 669
8.725.2.4 reconfigReqd . . . 669
8.725.2.5 sessionEndReason . . . 669
8.725.2.6 techName . . . 669
8.725.2.7 verboseSessnEndReason . . . 669
8.725.2.8 verboseSessnEndReasonType . . . 669

8.726 unpack_wds_SLQSSetWdsEventCallback_ind_t Struct Reference . . . 669
8.726.1 Detailed Description . . . 670
8.726.2 Field Documentation . . . 670
8.726.2.1 currDBTechAvail . . . 670
8.726.2.2 currNWInfo . . . 670
8.726.2.3 dataSysStatAvail . . . 670
8.726.2.4 dBTechAvail . . . 670
8.726.2.5 dBtechExtAvail . . . 670
8.726.2.6 dBTechExtRatValue . . . 670
8.726.2.7 dBTechExtSoMask . . . 670
8.726.2.8 dBTechnology . . . 670
8.726.2.9 dBtechnologyExt . . . 671
8.726.2.10 dormancyStatAvail . . . 671
8.726.2.11 dormancyStatus . . . 671
8.726.2.12 mipstatAvail . . . 671
8.726.2.13 mipStatus . . . 671
8.726.2.14 netInfoLen . . . 671
8.726.2.15 prefNetwork . . . 671
8.726.2.16 ratMask . . . 671
8.726.2.17 rx_bytes . . . 671
8.726.2.18 rx_pkts . . . 671
8.726.2.19 soMask . . . 671
8.726.2.20 tx_bytes . . . 671
8.726.2.21 tx_pkts . . . 671
8.726.2.22 xferStatAvail . . . 671
8.727 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 671
8.727.1 Detailed Description . . . 671
8.727.2 Field Documentation . . . 671
8.727.2.1 pHwConfig . . . 671
8.727.2.2 pRequestOptionList . . . 671
8.728 unpack_wds_SLQSSGetLoopback_t Struct Reference . . . 672
8.728.1 Detailed Description . . . 672
8.728.2 Field Documentation . . . 672
8.728.2.1 ByteLoopbackMode . . . 672
8.728.2.2 ByteLoopbackMultiplier . . . 672
8.729 unpack_wds_SLQSStartDataSession_t Struct Reference . . . 672
8.729.1 Detailed Description . . . 672
8.729.2 Field Documentation . . . 673
8.729.2.1 pFailureReason . . . 673
8.729.2.2 psid . . . 673

8.729.2.3 pVerboseFailReasonType . . . 673
8.729.2.4 pVerboseFailureReason . . . 673
8.730 unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 673
8.730.1 Detailed Description . . . 673
8.730.2 Field Documentation . . . 674
8.730.2.1 apnName . . . 674
8.730.2.2 bearerId . . . 674
8.730.2.3 contextId . . . 674
8.730.2.4 ipv4Address . . . 674
8.730.2.5 ipv4GWAddress . . . 675
8.730.2.6 ipv6Address . . . 675
8.730.2.7 ipv6GWAddress . . . 675
8.730.2.8 prDNSIPv4Address . . . 675
8.730.2.9 prDNSIPv6Address . . . 675
8.730.2.10 prPCSCFIPv4Address . . . 675
8.730.2.11 prPCSCFIPv6Address . . . 675
8.730.2.12 seDNSIPv4Address . . . 675
8.730.2.13 seDNSIPv6Address . . . 675
8.730.2.14 sePCSCFIPv4Address . . . 675
8.730.2.15 sePCSCFIPv6Address . . . 675
8.731 UnPackGetProfileSettingOut Struct Reference . . . 675
8.731.1 Detailed Description . . . 675
8.731.2 Field Documentation . . . 675
8.731.2.1 curProfile . . . 675
8.731.2.2 pExtErrCode . . . 675
8.732 unpackWdsProfileParam Union Reference . . . 675
8.732.1 Detailed Description . . . 676
8.732.2 Field Documentation . . . 676
8.732.2.1 SlqsProfile3GPP . . . 676
8.732.2.2 SlqsProfile3GPP2 . . . 676
8.733 voice_airTimer Struct Reference . . . 676
8.733.1 Detailed Description . . . 676
8.733.2 Field Documentation . . . 676
8.733.2.1 airTimerValue . . . 676
8.733.2.2 namID . . . 676
8.734 voice_allCallsAlphaIDInfo Struct Reference . . . 677
8.734.1 Detailed Description . . . 677
8.734.2 Field Documentation . . . 677
8.734.2.1 AlphaIDInfo . . . 677
8.734.2.2 callID . . . 677

8.735voice_allCallsDiagInfo Struct Reference . . . 677
8.735.1 Detailed Description . . . 677
8.735.2 Field Documentation . . . 677
8.735.2.1 callID . . . 677
8.735.2.2 DiagInfo . . . 677
8.736voice_allCallsUUSInfo Struct Reference . . . 678
8.736.1 Detailed Description . . . 678
8.736.2 Field Documentation . . . 678
8.736.2.1 callID . . . 678
8.736.2.2 uusInfo . . . 678
8.737voice_alphaIDInfo Struct Reference . . . 678
8.737.1 Detailed Description . . . 678
8.737.2 Field Documentation . . . 679
8.737.2.1 alphaDcs . . . 679
8.737.2.2 alphaLen . . . 679
8.737.2.3 alphaText . . . 679
8.738voice_arrAlertingPattern Struct Reference . . . 679
8.738.1 Detailed Description . . . 679
8.738.2 Field Documentation . . . 679
8.738.2.1 alertingPattern . . . 679
8.738.2.2 callID . . . 679
8.738.2.3 numInstances . . . 680
8.739voice_arrAlertingType Struct Reference . . . 680
8.739.1 Detailed Description . . . 680
8.739.2 Field Documentation . . . 680
8.739.2.1 AlertingType . . . 680
8.739.2.2 callID . . . 680
8.739.2.3 numInstances . . . 680
8.740voice_arrAlphaID Struct Reference . . . 680
8.740.1 Detailed Description . . . 680
8.740.2 Field Documentation . . . 681
8.740.2.1 allCallsAlphaIDInfoArr . . . 681
8.740.2.2 numInstances . . . 681
8.741voice_arrCalledPartyNum Struct Reference . . . 681
8.741.1 Detailed Description . . . 681
8.741.2 Field Documentation . . . 681
8.741.2.1 CalledPartyNum . . . 681
8.741.2.2 numInstances . . . 681
8.742voice_arrCallEndReason Struct Reference . . . 681
8.742.1 Detailed Description . . . 682

8.742.2 Field Documentation . . . 682
8.742.2.1 callEndReason . . . 682
8.742.2.2 callID . . . 682
8.742.2.3 numInstances . . . 682
8.743voice_arrCallInfo Struct Reference . . . 682
8.743.1 Detailed Description . . . 682
8.743.2 Field Documentation . . . 682
8.743.2.1 getAllCallInfo . . . 682
8.743.2.2 numInstances . . . 683
8.744voice_arrConnectPartyNum Struct Reference . . . 683
8.744.1 Detailed Description . . . 683
8.744.2 Field Documentation . . . 683
8.744.2.1 ConnectedPartyNum . . . 683
8.744.2.2 numInstances . . . 683
8.745voice_arrDiagInfo Struct Reference . . . 683
8.745.1 Detailed Description . . . 683
8.745.2 Field Documentation . . . 684
8.745.2.1 DiagInfo . . . 684
8.745.2.2 numInstances . . . 684
8.746voice_arrRedirPartyNum Struct Reference . . . 684
8.746.1 Detailed Description . . . 684
8.746.2 Field Documentation . . . 684
8.746.2.1 numInstances . . . 684
8.746.2.2 RedirPartyNum . . . 684
8.747voice_arrRemotePartyName Struct Reference . . . 684
8.747.1 Detailed Description . . . 684
8.747.2 Field Documentation . . . 685
8.747.2.1 GetAllCallRmtPtyName . . . 685
8.747.2.2 numInstances . . . 685
8.748voice_arrRemotePartyNum Struct Reference . . . 685
8.748.1 Detailed Description . . . 685
8.748.2 Field Documentation . . . 685
8.748.2.1 numInstances . . . 685
8.748.2.2 RmtPtyNum . . . 685
8.749voice_arrSvcOption Struct Reference . . . 685
8.749.1 Detailed Description . . . 685
8.749.2 Field Documentation . . . 686
8.749.2.1 callID . . . 686
8.749.2.2 numInstances . . . 686
8.749.2.3 srvOption . . . 686

8.750voice_arrUUSInfo Struct Reference 686
8.750.1 Detailed Description 686
8.750.2 Field Documentation 686
8.750.2.1 AllCallsUUSInfo 686
8.750.2.2 numInstances 686
8.751voice_burstDTMFInfo Struct Reference 686
8.751.1 Detailed Description 687
8.751.2 Field Documentation 687
8.751.2.1 digitCnt 687
8.751.2.2 pCallID 687
8.751.2.3 pDigitBuff 687
8.752voice_calledPartySubAdd Struct Reference 687
8.752.1 Detailed Description 687
8.752.2 Field Documentation 688
8.752.2.1 extBit 688
8.752.2.2 oddEvenInd 688
8.752.2.3 subAddr 688
8.752.2.4 subAddrLen 688
8.752.2.5 subAddrType 688
8.753voice_callFwdTypeAndPlan Struct Reference 688
8.753.1 Detailed Description 688
8.753.2 Field Documentation 689
8.753.2.1 numberPlan 689
8.753.2.2 numberType 689
8.754voice_callFWExtInfo Struct Reference 689
8.754.1 Detailed Description 689
8.754.2 Field Documentation 691
8.754.2.1 noReplyTimer 691
8.754.2.2 number 691
8.754.2.3 numLen 691
8.754.2.4 numPlan 691
8.754.2.5 numType 691
8.754.2.6 PI 691
8.754.2.7 SI 691
8.754.2.8 SvcClass 691
8.754.2.9 SvcStatus 691
8.755voice_callFWInfo Struct Reference 691
8.755.1 Detailed Description 691
8.755.2 Field Documentation 692
8.755.2.1 noReplyTimer 692

8.755.2.2 number . . . 692
8.755.2.3 numLen . . . 692
8.755.2.4 SvcClass . . . 692
8.755.2.5 SvcStatus . . . 692
8.756 voice_callInfo Struct Reference . . . 692
8.756.1 Detailed Description . . . 692
8.756.2 Field Documentation . . . 693
8.756.2.1 callID . . . 693
8.756.2.2 callState . . . 694
8.756.2.3 callType . . . 694
8.756.2.4 direction . . . 694
8.756.2.5 mode . . . 694
8.757 voice_ccSUPSType Struct Reference . . . 694
8.757.1 Detailed Description . . . 694
8.757.2 Field Documentation . . . 694
8.757.2.1 reason . . . 694
8.757.2.2 svcType . . . 694
8.758 voice_CLIPResp Struct Reference . . . 694
8.758.1 Detailed Description . . . 695
8.758.2 Field Documentation . . . 695
8.758.2.1 ActiveStatus . . . 695
8.758.2.2 ProvisionStatus . . . 695
8.759 voice_CLIRResp Struct Reference . . . 695
8.759.1 Detailed Description . . . 695
8.759.2 Field Documentation . . . 696
8.759.2.1 ActiveStatus . . . 696
8.759.2.2 ProvisionStatus . . . 696
8.760 voice_CNAPResp Struct Reference . . . 696
8.760.1 Detailed Description . . . 696
8.760.2 Field Documentation . . . 696
8.760.2.1 ActiveStatus . . . 696
8.760.2.2 ProvisionStatus . . . 696
8.761 voice_COLPResp Struct Reference . . . 697
8.761.1 Detailed Description . . . 697
8.761.2 Field Documentation . . . 697
8.761.2.1 ActiveStatus . . . 697
8.761.2.2 ProvisionStatus . . . 697
8.762 voice_COLRResp Struct Reference . . . 697
8.762.1 Detailed Description . . . 697
8.762.2 Field Documentation . . . 698

8.762.2.1 ActiveStatus . . . 698
8.762.2.2 ProvisionStatus . . . 698
8.763 voice_connectNumInfo Struct Reference . . . 698
8.763.1 Detailed Description . . . 698
8.763.2 Field Documentation . . . 699
8.763.2.1 callerID . . . 699
8.763.2.2 callerIDLen . . . 699
8.763.2.3 numPlan . . . 699
8.763.2.4 numPresInd . . . 700
8.763.2.5 numType . . . 700
8.763.2.6 screeningInd . . . 700
8.764 voice_CUGInfo Struct Reference . . . 700
8.764.1 Detailed Description . . . 700
8.764.2 Field Documentation . . . 700
8.764.2.1 CUGIndex . . . 700
8.764.2.2 SuppOA . . . 700
8.764.2.3 SuppPrefCUG . . . 700
8.765 voice_curAMRConfig Struct Reference . . . 700
8.765.1 Detailed Description . . . 700
8.765.2 Field Documentation . . . 701
8.765.2.1 gsmAmrStat . . . 701
8.765.2.2 wcdmaAmrStat . . . 701
8.766 voice_diagInfo Struct Reference . . . 701
8.766.1 Detailed Description . . . 701
8.766.2 Field Documentation . . . 701
8.766.2.1 diagInfoLen . . . 701
8.766.2.2 diagnosticInfo . . . 701
8.767 voice_DTMFLengths Struct Reference . . . 702
8.767.1 Detailed Description . . . 702
8.767.2 Field Documentation . . . 702
8.767.2.1 DTMFInterdigitInterval . . . 702
8.767.2.2 DTMFPulseWidth . . . 702
8.768 voice_getAllCallInformation Struct Reference . . . 702
8.768.1 Detailed Description . . . 702
8.768.2 Field Documentation . . . 703
8.768.2.1 ALS . . . 703
8.768.2.2 CallInfo . . . 703
8.768.2.3 isEmpty . . . 703
8.769 voice_getAllCallRmtPtyName Struct Reference . . . 703
8.769.1 Detailed Description . . . 703

8.769.2 Field Documentation . . . 703
8.769.2.1 callID . . . 703
8.769.2.2 RemotePartyName . . . 703
8.770 voice_getAllCallRmtPtyNum Struct Reference . . . 703
8.770.1 Detailed Description . . . 704
8.770.2 Field Documentation . . . 704
8.770.2.1 callID . . . 704
8.770.2.2 RemotePartyNum . . . 704
8.771 voice_getCallFWExtInfo Struct Reference . . . 704
8.771.1 Detailed Description . . . 704
8.771.2 Field Documentation . . . 704
8.771.2.1 CallFWExtInfo . . . 704
8.771.2.2 numInstances . . . 704
8.772 voice_getCallFWInfo Struct Reference . . . 705
8.772.1 Detailed Description . . . 705
8.772.2 Field Documentation . . . 705
8.772.2.1 CallFWInfo . . . 705
8.772.2.2 numInstances . . . 705
8.773 voice_peerNumberInfo Struct Reference . . . 705
8.773.1 Detailed Description . . . 705
8.773.2 Field Documentation . . . 707
8.773.2.1 callID . . . 707
8.773.2.2 number . . . 707
8.773.2.3 numLen . . . 707
8.773.2.4 numPI . . . 707
8.773.2.5 numPlan . . . 707
8.773.2.6 numSI . . . 707
8.773.2.7 numType . . . 707
8.774 voice_prefVoiceSO Struct Reference . . . 707
8.774.1 Detailed Description . . . 707
8.774.2 Field Documentation . . . 709
8.774.2.1 evrcCapability . . . 709
8.774.2.2 homeOrigVoiceSO . . . 709
8.774.2.3 homePageVoiceSO . . . 709
8.774.2.4 namID . . . 709
8.774.2.5 roamOrigVoiceSO . . . 709
8.775 voice_remotePartyName Struct Reference . . . 709
8.775.1 Detailed Description . . . 709
8.775.2 Field Documentation . . . 709
8.775.2.1 callerName . . . 710

8.775.2.2 codingScheme . . . 710
8.775.2.3 nameLen . . . 710
8.775.2.4 namePI . . . 710
8.776 voice_remotePartyNum Struct Reference . . . 710
8.776.1 Detailed Description . . . 710
8.776.2 Field Documentation . . . 710
8.776.2.1 numLen . . . 710
8.776.2.2 presentationInd . . . 710
8.776.2.3 remPartyNumber . . . 710
8.777 voice_roamTimer Struct Reference . . . 710
8.777.1 Detailed Description . . . 711
8.777.2 Field Documentation . . . 711
8.777.2.1 namID . . . 711
8.777.2.2 roamTimerValue . . . 711
8.778 voice_USSInfo Struct Reference . . . 711
8.778.1 Detailed Description . . . 711
8.778.2 Field Documentation . . . 711
8.778.2.1 ussData . . . 712
8.778.2.2 ussDCS . . . 712
8.778.2.3 ussLen . . . 712
8.779 voice_UUSInfo Struct Reference . . . 712
8.779.1 Detailed Description . . . 712
8.779.2 Field Documentation . . . 713
8.779.2.1 UUSData . . . 713
8.779.2.2 UUSDatalen . . . 713
8.779.2.3 UUSDcs . . . 713
8.779.2.4 UUSType . . . 713
8.780 wds_currNetworkInfo Struct Reference . . . 713
8.780.1 Detailed Description . . . 713
8.780.2 Field Documentation . . . 714
8.780.2.1 NetworkType . . . 714
8.780.2.2 RATMask . . . 714
8.780.2.3 SOMask . . . 714
8.781 wds_DataULongLongTlv Struct Reference . . . 714
8.781.1 Detailed Description . . . 714
8.781.2 Field Documentation . . . 714
8.781.2.1 TlvPresent . . . 714
8.781.2.2 ullData . . . 715
8.782 wds_DataULongTlv Struct Reference . . . 715
8.782.1 Detailed Description . . . 715

8.782.2 Field Documentation . . . 715
8.782.2.1 TlvPresent . . . 715
8.782.2.2 ulData . . . 715
8.783 wds_DHCPLeaseOptTlv Struct Reference . . . 715
8.783.1 Detailed Description . . . 715
8.783.2 Field Documentation . . . 715
8.783.2.1 numOpt . . . 715
8.783.2.2 optList . . . 715
8.783.2.3 optListData . . . 715
8.783.2.4 TlvPresent . . . 715
8.784 wds_DHCPLeaseStateTlv Struct Reference . . . 716
8.784.1 Detailed Description . . . 716
8.784.2 Field Documentation . . . 716
8.784.2.1 leaseState . . . 716
8.784.2.2 TlvPresent . . . 716
8.785 wds_DHCPOpt Struct Reference . . . 716
8.785.1 Detailed Description . . . 716
8.785.2 Field Documentation . . . 717
8.785.2.1 optCode . . . 717
8.785.2.2 optValLen . . . 717
8.785.2.3 pOptVal . . . 717
8.786 wds_DHCPProfileIdTlv Struct Reference . . . 717
8.786.1 Detailed Description . . . 717
8.786.2 Field Documentation . . . 717
8.786.2.1 profileId . . . 717
8.786.2.2 profileType . . . 717
8.786.2.3 TlvPresent . . . 717
8.787 wds_DHCPv4HWConfig Struct Reference . . . 717
8.787.1 Detailed Description . . . 717
8.787.2 Field Documentation . . . 718
8.787.2.1 chaddr . . . 718
8.787.2.2 chaddrLen . . . 718
8.787.2.3 hwType . . . 718
8.788 wds_DHCPv4Option Struct Reference . . . 718
8.788.1 Detailed Description . . . 718
8.788.2 Field Documentation . . . 718
8.788.2.1 optCode . . . 718
8.788.2.2 optVal . . . 718
8.788.2.3 optValLen . . . 718
8.789 wds_DHCPv4OptionList Struct Reference . . . 719

8.789.1 Detailed Description . . . 719
8.789.2 Field Documentation . . . 719
8.789.2.1 numOpt . . . 719
8.789.2.2 pOptList . . . 719
8.790 wds_DHCPv4ProfileId Struct Reference . . . 719
8.790.1 Detailed Description . . . 719
8.790.2 Field Documentation . . . 719
8.790.2.1 profileId . . . 719
8.790.2.2 profileType . . . 719
8.791 wds_Domain Struct Reference . . . 720
8.791.1 Detailed Description . . . 720
8.791.2 Field Documentation . . . 720
8.791.2.1 domainLen . . . 720
8.791.2.2 domainName . . . 720
8.792 wds_DomainNameList Struct Reference . . . 720
8.792.1 Detailed Description . . . 720
8.792.2 Field Documentation . . . 720
8.792.2.1 domain . . . 720
8.792.2.2 numInstances . . . 720
8.793 wds_GPRSQoS Struct Reference . . . 721
8.793.1 Detailed Description . . . 721
8.793.2 Field Documentation . . . 721
8.793.2.1 delayClass . . . 721
8.793.2.2 meanThroughputClass . . . 721
8.793.2.3 peakThroughputClass . . . 721
8.793.2.4 precedenceClass . . . 721
8.793.2.5 reliabilityClass . . . 721
8.794 wds_IPv4AdTlv Struct Reference . . . 721
8.794.1 Detailed Description . . . 722
8.794.2 Field Documentation . . . 722
8.794.2.1 IPv4Addr . . . 722
8.794.2.2 TlvPresent . . . 722
8.795 wds_IPV6AddressInfo Struct Reference . . . 722
8.795.1 Detailed Description . . . 722
8.795.2 Field Documentation . . . 722
8.795.2.1 IPAddressV6 . . . 722
8.795.2.2 IPV6PrefixLen . . . 722
8.796 wds_IPV6GWAddressInfo Struct Reference . . . 722
8.796.1 Detailed Description . . . 722
8.796.2 Field Documentation . . . 723

8.796.2.1 gwAddressV6 . . . 723
8.796.2.2 gwV6PrefixLen . . . 723
8.797 wds_PCSCFFQDNAddress Struct Reference . . . 723
8.797.1 Detailed Description . . . 723
8.797.2 Field Documentation . . . 723
8.797.2.1 fqdnAddr . . . 723
8.797.2.2 fqdnLen . . . 723
8.798 wds_PCSCFFQDNAddressList Struct Reference . . . 723
8.798.1 Detailed Description . . . 723
8.798.2 Field Documentation . . . 724
8.798.2.1 numInstances . . . 724
8.798.2.2 pcsfFQDNAddress . . . 724
8.799 wds_PCSCFIPv4ServerAddressList Struct Reference . . . 724
8.799.1 Detailed Description . . . 724
8.799.2 Field Documentation . . . 724
8.799.2.1 numInstances . . . 724
8.799.2.2 pscsfIPv4Addr . . . 724
8.800 wds_ProfileIdentifier Struct Reference . . . 724
8.800.1 Detailed Description . . . 724
8.800.2 Field Documentation . . . 725
8.800.2.1 profileIndex . . . 725
8.800.2.2 profileType . . . 725
8.801 wds_profileInfo Union Reference . . . 725
8.801.1 Detailed Description . . . 725
8.801.2 Field Documentation . . . 725
8.801.2.1 SlqsProfile3GPP . . . 725
8.801.2.2 SlqsProfile3GPP2 . . . 725
8.802 wds_TrStatInd Struct Reference . . . 725
8.802.1 Detailed Description . . . 725
8.802.2 Field Documentation . . . 726
8.802.2.1 statsMask . . . 726
8.802.2.2 statsPeriod . . . 726
8.803 wds_UMTSMinQoS Struct Reference . . . 726
8.803.1 Detailed Description . . . 726
8.803.2 Field Documentation . . . 728
8.803.2.1 deliveryErrSDU . . . 728
8.803.2.2 grntDownlinkBitrate . . . 728
8.803.2.3 grntUplinkBitrate . . . 728
8.803.2.4 maxDownlinkBitrate . . . 728
8.803.2.5 maxSDUSize . . . 728

8.803.2.6 maxUplinkBitrate . . . 728
8.803.2.7 qosDeliveryOrder . . . 728
8.803.2.8 resBerRatio . . . 728
8.803.2.9 sduErrorRatio . . . 728
8.803.2.10 trafficClass . . . 728
8.803.2.11 trafficPriority . . . 728
8.803.2.12 transferDelay . . . 728
8.804 wdsDhcpv4HwConfig Struct Reference . . . 728
8.804.1 Detailed Description . . . 729
8.804.2 Field Documentation . . . 729
8.804.2.1 chaddr . . . 729
8.804.2.2 chaddrLen . . . 729
8.804.2.3 hwType . . . 729
8.805 wdsDhcpv4Option Struct Reference . . . 729
8.805.1 Detailed Description . . . 729
8.805.2 Field Documentation . . . 729
8.805.2.1 optCode . . . 729
8.805.2.2 optVal . . . 729
8.805.2.3 optValLen . . . 730
8.806 wdsDhcpv4OptionList Struct Reference . . . 730
8.806.1 Detailed Description . . . 730
8.806.2 Field Documentation . . . 730
8.806.2.1 numOpt . . . 730
8.806.2.2 pOptList . . . 730
8.807 wdsDhcpv4ProfileId Struct Reference . . . 730
8.807.1 Detailed Description . . . 730
8.807.2 Field Documentation . . . 730
8.807.2.1 profileId . . . 730
8.807.2.2 profileType . . . 730

9 File Documentation **731**

9.1 apdoxypages.c File Reference . . . 731
9.1.1 Detailed Description . . . 731
9.2 cat.h File Reference . . . 731
9.2.1 Macro Definition Documentation . . . 732
9.2.1.1 CAN_COMMON_EVENT_TLV_NUMBER . . . 732
9.2.1.2 CAT_EVENT_DATA_MAX_LENGTH . . . 732
9.2.2 Function Documentation . . . 732
9.2.2.1 pack_cat_CATSendEnvelopeCommand . . . 732
9.2.2.2 pack_cat_CATSendTerminalResponse . . . 732

9.2.2.3 pack_cat_SetCATEventCallback . . . 733
9.2.2.4 unpack_cat_CATSendEnvelopeCommand . . . 733
9.2.2.5 unpack_cat_CATSendTerminalResponse . . . 733
9.2.2.6 unpack_cat_SetCATEventCallback . . . 734
9.2.2.7 unpack_cat_SetCatEventCallback_ind . . . 734
9.3 common.h File Reference . . . 734
9.3.1 Macro Definition Documentation . . . 736
9.3.1.1 DEAULT_LOC_TIMEOUT_IN_SEC . . . 736
9.3.1.2 MINREQBKLEN . . . 736
9.3.1.3 MSGID_AND_LEN . . . 736
9.3.1.4 MSGID_DONT_CARE . . . 736
9.3.1.5 SDK_VALIDATE_INPUT_PACK_PARAM . . . 736
9.3.1.6 SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID . . . 736
9.3.1.7 SDU_HDR_LEN . . . 737
9.3.1.8 UNUSEDPARAM . . . 737
9.3.2 Typedef Documentation . . . 737
9.3.2.1 logger . . . 737
9.3.3 Enumeration Type Documentation . . . 737
9.3.3.1 eLOG_LEVEL . . . 737
9.3.3.2 eQMI_SVC . . . 737
9.3.3.3 eTimeout . . . 738
9.3.3.4 msgtype . . . 738
9.3.4 Function Documentation . . . 738
9.3.4.1 fill_pack_ctx . . . 738
9.3.4.2 fill_sdu_hdr . . . 738
9.3.4.3 get_version . . . 738
9.3.4.4 helper_get_resp_ctx . . . 738
9.3.4.5 helper_get_xid . . . 739
9.3.4.6 helper_set_log_func . . . 739
9.3.4.7 helper_set_log_lvl . . . 739
9.3.4.8 liteqmi_GetVersion . . . 739
9.3.4.9 liteqmi_helper_decode7bitAsciiEncString . . . 739
9.3.4.10 liteqmi_log . . . 739
9.3.4.11 unpack_result_code_only . . . 739
9.3.5 Variable Documentation . . . 739
9.3.5.1 glog . . . 739
9.3.5.2 gloglvl . . . 739
9.4 dms.h File Reference . . . 739
9.4.1 Macro Definition Documentation . . . 746
9.4.1.1 ACT_CODE_MAX_SIZE . . . 746

9.4.1.2 CK_MAX_SIZE . . . 746
9.4.1.3 DMS_IMGDETAILS_LEN . . . 746
9.4.1.4 DMS_MAX_CUST_ID_LEN . . . 746
9.4.1.5 DMS_MAX_CUST_VALUE_LEN . . . 746
9.4.1.6 DMS_MAX_FWUPDATE_LOG_STR_SZ . . . 746
9.4.1.7 DMS_MAX_FWUPDATE_REF_STR_SZ . . . 746
9.4.1.8 DMS_PM_FACTORY . . . 746
9.4.1.9 DMS_PM_LOW . . . 746
9.4.1.10 DMS_PM_OFFLINE . . . 746
9.4.1.11 DMS_PM_ONLINE . . . 746
9.4.1.12 DMS_PM_PERSISTENT_LOW . . . 746
9.4.1.13 DMS_PM_RESET . . . 746
9.4.1.14 DMS_PM_SHUT_DOWN . . . 746
9.4.1.15 DMS_SET_REPORT_DISABLE . . . 746
9.4.1.16 DMS_SET_REPORT_ENABLE . . . 746
9.4.1.17 DMS_SLQSFWINFO_APPVERSION_SZ . . . 746
9.4.1.18 DMS_SLQSFWINFO_BOOTVERSION_SZ . . . 746
9.4.1.19 DMS_SLQSFWINFO_CARRIER_SZ . . . 746
9.4.1.20 DMS_SLQSFWINFO_CUR_CARR_NAME . . . 746
9.4.1.21 DMS_SLQSFWINFO_CUR_CARR_REV . . . 746
9.4.1.22 DMS_SLQSFWINFO_MODELID_SZ . . . 746
9.4.1.23 DMS_SLQSFWINFO_PACKAGEID_SZ . . . 746
9.4.1.24 DMS_SLQSFWINFO_PRIVERSION_SZ . . . 746
9.4.1.25 DMS_SLQSFWINFO_SKU_SZ . . . 746
9.4.1.26 DMS_SWI_SET_IND_DISABLE . . . 746
9.4.1.27 DMS_SWI_SET_IND_ENABLE . . . 746
9.4.1.28 DMS_UINT8_MAX_STRING_SZ . . . 747
9.4.1.29 ERI_DATA_MAX_SIZE . . . 747
9.4.1.30 MAX_BUILD_ID_LEN . . . 747
9.4.1.31 MEID_MAX_SIZE . . . 747
9.4.1.32 SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH . . . 747
9.4.1.33 SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH . . . 747
9.4.1.34 SPC_SIZE . . . 747
9.4.1.35 UNIQUE_ID_LEN . . . 747
9.4.2 Function Documentation . . . 747
9.4.2.1 pack_dms_ActivateAutomatic . . . 747
9.4.2.2 pack_dms_GetActivationState . . . 747
9.4.2.3 pack_dms_GetBandCapability . . . 748
9.4.2.4 pack_dms_GetCrashAction . . . 748
9.4.2.5 pack_dms_GetCustFeature . . . 748

9.4.2.6 pack_dms_GetCustFeaturesV2 . . . 749
9.4.2.7 pack_dms_GetDeviceCap . . . 749
9.4.2.8 pack_dms_GetDeviceCapabilities . . . 749
9.4.2.9 pack_dms_GetDeviceHardwareRev . . . 750
9.4.2.10 pack_dms_GetDeviceMfr . . . 750
9.4.2.11 pack_dms_GetDeviceSerialNumbers . . . 751
9.4.2.12 pack_dms_GetFirmwareInfo . . . 751
9.4.2.13 pack_dms_GetFirmwareRevision . . . 751
9.4.2.14 pack_dms_GetFirmwareRevisions . . . 752
9.4.2.15 pack_dms_GetFSN . . . 752
9.4.2.16 pack_dms_GetHardwareRevision . . . 752
9.4.2.17 pack_dms_GetIMSI . . . 753
9.4.2.18 pack_dms_GetManufacturer . . . 753
9.4.2.19 pack_dms_GetModelID . . . 753
9.4.2.20 pack_dms_GetNetworkTime . . . 754
9.4.2.21 pack_dms_GetOfflineReason . . . 754
9.4.2.22 pack_dms_GetPower . . . 754
9.4.2.23 pack_dms_GetPRLVersion . . . 755
9.4.2.24 pack_dms_GetSerialNumbers . . . 755
9.4.2.25 pack_dms_GetUSBComp . . . 756
9.4.2.26 pack_dms_GetVoiceNumber . . . 756
9.4.2.27 pack_dms_ResetToFactoryDefaults . . . 756
9.4.2.28 pack_dms_SetActivationStatusCallback . . . 757
9.4.2.29 pack_dms_SetCrashAction . . . 757
9.4.2.30 pack_dms_SetCustFeature . . . 757
9.4.2.31 pack_dms_SetCustFeaturesV2 . . . 758
9.4.2.32 pack_dms_SetEventReport . . . 758
9.4.2.33 pack_dms_SetFirmwarePreference . . . 758
9.4.2.34 pack_dms_SetPower . . . 759
9.4.2.35 pack_dms_SetUSBComp . . . 759
9.4.2.36 pack_dms_SLQSDmsSwiGetResetInfo . . . 760
9.4.2.37 pack_dms_SLQSDmsSwiIndicationRegister . . . 760
9.4.2.38 pack_dms_SLQSGetBandCapability . . . 760
9.4.2.39 pack_dms_SLQSGetERIFile . . . 761
9.4.2.40 pack_dms_SLQSSwiClearDyingGaspStatistics . . . 761
9.4.2.41 pack_dms_SLQSSwiGetCrashInfo . . . 761
9.4.2.42 pack_dms_SLQSSwiGetDyingGaspCfg . . . 762
9.4.2.43 pack_dms_SLQSSwiGetDyingGaspStatistics . . . 762
9.4.2.44 pack_dms_SLQSSwiGetFirmwareCurr . . . 762
9.4.2.45 pack_dms_SLQSSwiGetFwUpdateStatus . . . 763

9.4.2.46 pack_dms_SLQSSwiGetHostDevInfo . . . 763
9.4.2.47 pack_dms_SLQSSwiGetOSInfo . . . 763
9.4.2.48 pack_dms_SLQSSwiGetSerialNoExt . . . 764
9.4.2.49 pack_dms_SLQSSwiSetDyingGaspCfg . . . 764
9.4.2.50 pack_dms_SLQSSwiSetHostDevInfo . . . 764
9.4.2.51 pack_dms_SLQSSwiSetOSInfo . . . 765
9.4.2.52 pack_dms_SLQSUIMGetState . . . 765
9.4.2.53 pack_dms_UIMChangePIN . . . 766
9.4.2.54 pack_dms_UIMGetControlKeyStatus . . . 766
9.4.2.55 pack_dms_UIMGetICCID . . . 766
9.4.2.56 pack_dms_UIMGetPINStatus . . . 767
9.4.2.57 pack_dms_UIMSetControlKeyProtection . . . 767
9.4.2.58 pack_dms_UIMSetPINProtection . . . 767
9.4.2.59 pack_dms_UIMUnblockControlKey . . . 768
9.4.2.60 pack_dms_UIMUnblockPIN . . . 768
9.4.2.61 pack_dms_UIMVerifyPIN . . . 769
9.4.2.62 pack_dms_ValidateSPC . . . 769
9.4.2.63 unpack_dms_ActivateAutomatic . . . 769
9.4.2.64 unpack_dms_GetActivationState . . . 770
9.4.2.65 unpack_dms_GetBandCapability . . . 770
9.4.2.66 unpack_dms_GetCrashAction . . . 770
9.4.2.67 unpack_dms_GetCustFeature . . . 771
9.4.2.68 unpack_dms_GetCustFeaturesV2 . . . 771
9.4.2.69 unpack_dms_GetDeviceCap . . . 771
9.4.2.70 unpack_dms_GetDeviceCapabilities . . . 772
9.4.2.71 unpack_dms_GetDeviceHardwareRev . . . 772
9.4.2.72 unpack_dms_GetDeviceMfr . . . 772
9.4.2.73 unpack_dms_GetDeviceSerialNumbers . . . 773
9.4.2.74 unpack_dms_GetFirmwareInfo . . . 773
9.4.2.75 unpack_dms_GetFirmwareRevision . . . 773
9.4.2.76 unpack_dms_GetFirmwareRevisions . . . 774
9.4.2.77 unpack_dms_GetFSN . . . 774
9.4.2.78 unpack_dms_GetHardwareRevision . . . 774
9.4.2.79 unpack_dms_GetIMSI . . . 775
9.4.2.80 unpack_dms_GetManufacturer . . . 775
9.4.2.81 unpack_dms_GetModelID . . . 775
9.4.2.82 unpack_dms_GetNetworkTime . . . 776
9.4.2.83 unpack_dms_GetOfflineReason . . . 776
9.4.2.84 unpack_dms_GetPower . . . 776
9.4.2.85 unpack_dms_GetPRLVersion . . . 777

9.4.2.86 unpack_dms_GetSerialNumbers . . . 777
9.4.2.87 unpack_dms_GetUSBComp . . . 777
9.4.2.88 unpack_dms_GetVoiceNumber . . . 778
9.4.2.89 unpack_dms_ResetToFactoryDefaults . . . 778
9.4.2.90 unpack_dms_SetActivationStatusCallback . . . 778
9.4.2.91 unpack_dms_SetCrashAction . . . 779
9.4.2.92 unpack_dms_SetCustFeature . . . 779
9.4.2.93 unpack_dms_SetCustFeaturesV2 . . . 779
9.4.2.94 unpack_dms_SetEventReport . . . 780
9.4.2.95 unpack_dms_SetEventReport_ind . . . 780
9.4.2.96 unpack_dms_SetFirmwarePreference . . . 781
9.4.2.97 unpack_dms_SetPower . . . 781
9.4.2.98 unpack_dms_SetUSBComp . . . 781
9.4.2.99 unpack_dms_SLQSDmsSwiGetResetInfo . . . 782
9.4.2.100 unpack_dms_SLQSDmsSwiGetResetInfo_Ind . . . 782
9.4.2.101 unpack_dms_SLQSDmsSwiIndicationRegister . . . 782
9.4.2.102 unpack_dms_SLQSGetBandCapability . . . 783
9.4.2.103 unpack_dms_SLQSGetERIFile . . . 783
9.4.2.104 unpack_dms_SLQSSwiClearDyingGaspStatistics . . . 783
9.4.2.105 unpack_dms_SLQSSwiGetCrashInfo . . . 784
9.4.2.106 unpack_dms_SLQSSwiGetDyingGaspCfg . . . 784
9.4.2.107 unpack_dms_SLQSSwiGetDyingGaspStatistics . . . 785
9.4.2.108 unpack_dms_SLQSSwiGetFirmwareCurr . . . 785
9.4.2.109 unpack_dms_SLQSSwiGetFwUpdateStatus . . . 785
9.4.2.110 unpack_dms_SLQSSwiGetHostDevInfo . . . 786
9.4.2.111 unpack_dms_SLQSSwiGetOSInfo . . . 786
9.4.2.112 unpack_dms_SLQSSwiGetSerialNoExt . . . 786
9.4.2.113 unpack_dms_SLQSSwiSetDyingGaspCfg . . . 787
9.4.2.114 unpack_dms_SLQSSwiSetHostDevInfo . . . 787
9.4.2.115 unpack_dms_SLQSSwiSetOSInfo . . . 787
9.4.2.116 unpack_dms_SLQSUIMGetState . . . 788
9.4.2.117 unpack_dms_UIMChangePIN . . . 788
9.4.2.118 unpack_dms_UIMGetControlKeyStatus . . . 788
9.4.2.119 unpack_dms_UIMGetICCID . . . 789
9.4.2.120 unpack_dms_UIMGetPINStatus . . . 789
9.4.2.121 unpack_dms_UIMSetControlKeyProtection . . . 789
9.4.2.122 unpack_dms_UIMSetPINProtection . . . 790
9.4.2.123 unpack_dms_UIMUnblockControlKey . . . 790
9.4.2.124 unpack_dms_UIMUnblockPIN . . . 791
9.4.2.125 unpack_dms_UIMVerifyPIN . . . 791

9.4.2.126 unpack_dms_ValidateSPC . . . 791
9.5 fms.h File Reference . . . 792
9.5.1 Macro Definition Documentation . . . 793
9.5.1.1 FMS_FW_PRI_BUILD_MATCH_LEN . . . 793
9.5.1.2 FMS_GOBI_LISTENTRIES_MAX . . . 793
9.5.1.3 FMS_GOBI_MBN_BUILD_ID_STR_LEN . . . 793
9.5.1.4 FMS_GOBI_MBN_IMG_ID_STR_LEN . . . 793
9.5.1.5 FMS_IMAGE_ID_BUILD_ID_LEN . . . 793
9.5.1.6 FMS_IMAGE_ID_IMG_ID_LEN . . . 793
9.5.1.7 FMS_IMAGE_ID_MAX_ENTRIES . . . 793
9.5.1.8 FMS_IMAGE_ID_PRI_IMGTYPE . . . 793
9.5.1.9 FMS_MAX_IMAGE_ID_ELEMENT . . . 793
9.5.1.10 FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE . . . 793
9.5.2 Function Documentation . . . 793
9.5.2.1 GetValidFwPriCombinations . . . 793
9.5.2.2 pack_fms_GetImagesPreference . . . 793
9.5.2.3 pack_fms_GetStoredImages . . . 794
9.5.2.4 pack_fms_SetImagesPreference . . . 794
9.5.2.5 unpack_fms_GetImagesPreference . . . 794
9.5.2.6 unpack_fms_GetStoredImages . . . 794
9.5.2.7 unpack_fms_SetImagesPreference . . . 795
9.6 lite-fw.h File Reference . . . 795
9.6.1 Detailed Description . . . 796
9.6.2 Macro Definition Documentation . . . 797
9.6.2.1 FIRMWARE_INFO_STRING_SIZE . . . 797
9.6.2.2 IMG_MASK_CLEAR . . . 797
9.6.2.3 IMG_MASK_GENERIC . . . 797
9.6.2.4 IMG_MASK_MDM . . . 797
9.6.2.5 IMG_MASK_PRI . . . 797
9.6.2.6 libSDP_BuildImagesPreferenceRequest . . . 797
9.6.2.7 libSDP_CalculateImageMask . . . 797
9.6.2.8 LIBSDP_CARRIER_PACKAGE_SKU . . . 797
9.6.2.9 libSDP_CheckValidFirmwareInfo . . . 797
9.6.2.10 libSDP_DownloadFW . . . 797
9.6.2.11 libSDP_ExtractFirmwareParametersByPath . . . 797
9.6.2.12 libSDP_getFileType . . . 797
9.6.2.13 libSDP_GetModelFamily . . . 797
9.6.2.14 libSDP_GetVersion . . . 797
9.6.2.15 libsdp_set_log_func . . . 797
9.6.2.16 libsdp_SetReadBlockSize . . . 797

9.6.2.17 LIBSDP_SKU_STRING_LENGTH . . . 797
9.6.2.18 LITEFW_CARRIER_PACKAGE_SKU . . . 797
9.6.2.19 LITEFW_SKU_STRING_LENGTH . . . 797
9.6.3 Typedef Documentation . . . 797
9.6.3.1 libSDP_FirmwareInfo . . . 797
9.6.3.2 libsdplogger . . . 797
9.6.3.3 litefw_FirmwareInfo . . . 797
9.6.3.4 litefwlogger . . . 798
9.6.4 Enumeration Type Documentation . . . 798
9.6.4.1 litefw_Fw_Type . . . 798
9.6.4.2 litefw_fwdwl_error_codes . . . 798
9.6.4.3 litefw_Models . . . 799
9.6.5 Function Documentation . . . 799
9.6.5.1 litefw_BuildImagesPreferenceRequest . . . 799
9.6.5.2 litefw_CalculateImageMask . . . 799
9.6.5.3 litefw_CheckValidFirmwareInfo . . . 800
9.6.5.4 litefw_DownloadFW . . . 800
9.6.5.5 litefw_ExtractFirmwareParametersByPath . . . 801
9.6.5.6 litefw_getFileType . . . 801
9.6.5.7 litefw_GetModelFamily . . . 801
9.6.5.8 litefw_GetVersion . . . 802
9.6.5.9 litefw_set_log_func . . . 802
9.6.5.10 litefw_SetReadBlockSize . . . 802
9.6.5.11 litefw_switch_9x07_to_downloadmode . . . 802
9.7 loc.h File Reference . . . 803
9.7.1 Macro Definition Documentation . . . 805
9.7.1.1 LOC_UINT8_MAX_STRING_SZ . . . 805
9.7.1.2 LOCEVENTMASKBATCHFULLNOTIFICATION . . . 805
9.7.1.3 LOCEVENTMASKENGINESTATE . . . 805
9.7.1.4 LOCEVENTMASKFIXSESSIONSTATE . . . 806
9.7.1.5 LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION . . . 806
9.7.1.6 LOCEVENTMASKGEOFENCEBREACHNOTIFICATION . . . 806
9.7.1.7 LOCEVENTMASKGEOFENCEGENALERT . . . 806
9.7.1.8 LOCEVENTMASKGNSSMEASUREMENTREPORT . . . 806
9.7.1.9 LOCEVENTMASKGNSSSVINFO . . . 806
9.7.1.10 LOCEVENTMASKINJECTPOSITIONREQ . . . 806
9.7.1.11 LOCEVENTMASKINJECTPREDICTEDORBITSREQ . . . 806
9.7.1.12 LOCEVENTMASKINJECTTIMEREQ . . . 806
9.7.1.13 LOCEVENTMASKINJECTWIFIAPDATAREQ . . . 806
9.7.1.14 LOCEVENTMASKINVALIDVALUE . . . 806

9.7.1.15 LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT . . . 807
9.7.1.16 LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ . . . 807
9.7.1.17 LOCEVENTMASKMOTIONDATACONTROL . . . 807
9.7.1.18 LOCEVENTMASKNIGEOFENCENOTIFICATION . . . 807
9.7.1.19 LOCEVENTMASKNINOTIFYVERIFYREQ . . . 807
9.7.1.20 LOCEVENTMASKNMEA . . . 807
9.7.1.21 LOCEVENTMASKPEDOMETERCONTROL . . . 807
9.7.1.22 LOCEVENTMASKPOSITIONREPORT . . . 807
9.7.1.23 LOCEVENTMASKSENSORSTREAMINGREADYSTATUS . . . 807
9.7.1.24 LOCEVENTMASKSETSPISTREAMINGREPORT . . . 807
9.7.1.25 LOCEVENTMASKTIMESYNCREQ . . . 808
9.7.1.26 LOCEVENTMASKVEHICLEDATAREADYSTATUS . . . 808
9.7.1.27 LOCEVENTMASKWIFIREQ . . . 808
9.7.1.28 MAX_LOC_NMEA_STR_LEN . . . 808
9.7.1.29 MAX_SENSOR_DATA_LEN . . . 808
9.7.1.30 MAX_TEMP_DATA_LEN . . . 808
9.7.2 Enumeration Type Documentation . . . 808
9.7.2.1 anonymous enum . . . 808
9.7.3 Function Documentation . . . 808
9.7.3.1 pack_loc_DeleteAssistData . . . 808
9.7.3.2 pack_loc_EventRegister . . . 809
9.7.3.3 pack_loc_SetExtPowerState . . . 809
9.7.3.4 pack_loc_SetOperationMode . . . 809
9.7.3.5 pack_loc_SLQSLOCGetBestAvailPos . . . 810
9.7.3.6 pack_loc_SLQSLOCGetServer . . . 810
9.7.3.7 pack_loc_SLQSLOCInjectPosition . . . 810
9.7.3.8 pack_loc_SLQSLOCInjectSensorData . . . 811
9.7.3.9 pack_loc_SLQSLOCInjectUTCTime . . . 811
9.7.3.10 pack_loc_SLQSLOCSetCradleMountConfig . . . 811
9.7.3.11 pack_loc_Start . . . 812
9.7.3.12 pack_loc_Stop . . . 812
9.7.3.13 unpack_loc_BestAvailPos_Ind . . . 813
9.7.3.14 unpack_loc_DeleteAssistData . . . 813
9.7.3.15 unpack_loc_DeleteAssistData_Ind . . . 813
9.7.3.16 unpack_loc_EngineState_Ind . . . 814
9.7.3.17 unpack_loc_EventNMEA_Ind . . . 814
9.7.3.18 unpack_loc_EventRegister . . . 814
9.7.3.19 unpack_loc_GetServer_Ind . . . 815
9.7.3.20 unpack_loc_GnssSvInfo_Ind . . . 815
9.7.3.21 unpack_loc_PositionRpt_Ind . . . 815

9.7.3.22 unpack_loc_SetExtPowerConfig_Ind . . . 816
9.7.3.23 unpack_loc_SetExtPowerState . . . 816
9.7.3.24 unpack_loc_SetOperationMode . . . 816
9.7.3.25 unpack_loc_SetOperationMode_Ind . . . 817
9.7.3.26 unpack_loc_SLQSLOCGetBestAvailPos . . . 817
9.7.3.27 unpack_loc_SLQSLOCGetServer . . . 817
9.7.3.28 unpack_loc_SLQSLOCInjectPosition . . . 818
9.7.3.29 unpack_loc_SLQSLOCInjectSensorData . . . 818
9.7.3.30 unpack_loc_SLQSLOCInjectUTCTime . . . 818
9.7.3.31 unpack_loc_SLQSLOCSetCradleMountConfig . . . 819
9.7.3.32 unpack_loc_Start . . . 819
9.7.3.33 unpack_loc_Stop . . . 819
9.8 nas.h File Reference . . . 819
9.8.1 Macro Definition Documentation . . . 827
9.8.1.1 NAS_IMSI_M_S1_LENGTH . . . 827
9.8.1.2 NAS_IMSI_M_S2_LENGTH . . . 827
9.8.1.3 NAS_MAX_DESCRIPTION_LENGTH . . . 827
9.8.1.4 NAS_MAX_NUM_NETWORKS . . . 827
9.8.1.5 NAS_MAX_SCC_RX_INFO_INSTANCES . . . 827
9.8.1.6 NAS_NAM_NAME_LENGTH . . . 827
9.8.1.7 NAS_OTA_MESSAGE_MAX_BUF_SIZE . . . 827
9.8.1.8 NAS_PLMN_LENGTH . . . 827
9.8.1.9 NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST . . . 827
9.8.2 Enumeration Type Documentation . . . 827
9.8.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB . . . 828
9.8.2.2 LITEQMI_NAS_LTE_CPHY_SCELL_STATE . . . 828
9.8.2.3 NAS_LTE_CPHY_CA_BW_NRB_LITE . . . 828
9.8.2.4 NAS_LTE_CPHY_SCELL_STATE_LITE . . . 828
9.8.3 Function Documentation . . . 828
9.8.3.1 pack_nas_GetACCOLC . . . 828
9.8.3.2 pack_nas_GetANAAAAuthenticationStatus . . . 829
9.8.3.3 pack_nas_GetCDMANetworkParameters . . . 829
9.8.3.4 pack_nas_GetHomeNetwork . . . 829
9.8.3.5 pack_nas_GetHomeNetwork3GPP2 . . . 830
9.8.3.6 pack_nas_GetNetworkPreference . . . 830
9.8.3.7 pack_nas_GetRFInfo . . . 830
9.8.3.8 pack_nas_GetServingNetwork . . . 830
9.8.3.9 pack_nas_GetServingNetworkCapabilities . . . 831
9.8.3.10 pack_nas_GetSignalStrengths . . . 831
9.8.3.11 pack_nas_InitiateDomainAttach . . . 831

9.8.3.12 pack_nas_PerformNetworkScan . . . 832
9.8.3.13 pack_nas_SetACCOLC . . . 832
9.8.3.14 pack_nas_SetCDMANetworkParameters . . . 832
9.8.3.15 pack_nas_SetLURejectCallback . . . 833
9.8.3.16 pack_nas_SetNetworkPreference . . . 833
9.8.3.17 pack_nas_SetRFInfoCallback . . . 833
9.8.3.18 pack_nas_SLQSConfigSigInfo . . . 834
9.8.3.19 pack_nas_SLQSGetErrorRate . . . 834
9.8.3.20 pack_nas_SlqsGetLTECphyCAInfo . . . 834
9.8.3.21 pack_nas_SLQSGetNetworkTime . . . 835
9.8.3.22 pack_nas_SLQSGetOperatorNameData . . . 835
9.8.3.23 pack_nas_SLQSGetPLMNName . . . 835
9.8.3.24 pack_nas_SLQSGetServingSystem . . . 835
9.8.3.25 pack_nas_SLQSGetSignalStrength . . . 836
9.8.3.26 pack_nas_SLQSGetSysInfo . . . 836
9.8.3.27 pack_nas_SLQSGetSysSelectionPref . . . 836
9.8.3.28 pack_nas_SLQSInitiateNetworkRegistration . . . 837
9.8.3.29 pack_nas_SLQSNasConfigSigInfo2 . . . 837
9.8.3.30 pack_nas_SLQSNasGet3GPP2Subscription . . . 837
9.8.3.31 pack_nas_SLQSNasGetCellLocationInfo . . . 838
9.8.3.32 pack_nas_SLQSNasGetHDRColorCode . . . 838
9.8.3.33 pack_nas_SLQSNasGetSigInfo . . . 838
9.8.3.34 pack_nas_SLQSNasGetTxRxInfo . . . 839
9.8.3.35 pack_nas_SLQSNasIndicationRegisterExt . . . 839
9.8.3.36 pack_nas_SLQSNASSwiGetChannelLock . . . 840
9.8.3.37 pack_nas_SLQSNasSwiIndicationRegister . . . 840
9.8.3.38 pack_nas_SLQSNasSwiModemStatus . . . 840
9.8.3.39 pack_nas_SLQSNASSwiSetChannelLock . . . 841
9.8.3.40 pack_nas_SLQSSetBandPreference . . . 841
9.8.3.41 pack_nas_SLQSSetSignalStrengthsCallback . . . 841
9.8.3.42 pack_nas_SLQSSetSysSelectionPref . . . 842
9.8.3.43 pack_nas_SLQSSwiGetHDRPersonality . . . 842
9.8.3.44 pack_nas_SLQSSwiGetHDRProtSubtype . . . 842
9.8.3.45 pack_nas_SLQSSwiGetHRPDStats . . . 843
9.8.3.46 pack_nas_SLQSSwiGetLteCQI . . . 843
9.8.3.47 pack_nas_SLQSSwiGetLteSccRxInfo . . . 843
9.8.3.48 pack_nas_SLQSSwiNetworkDebug . . . 844
9.8.3.49 pack_nas_SLQSSwiPSDetach . . . 844
9.8.3.50 unpack_nas_GetACCOLC . . . 844
9.8.3.51 unpack_nas_GetANAAAAuthenticationStatus . . . 845

9.8.3.52 unpack_nas_GetCDMANetworkParameters . . . 845
9.8.3.53 unpack_nas_GetHomeNetwork . . . 845
9.8.3.54 unpack_nas_GetHomeNetwork3GPP2 . . . 846
9.8.3.55 unpack_nas_GetNetworkPreference . . . 846
9.8.3.56 unpack_nas_GetRFInfo . . . 846
9.8.3.57 unpack_nas_GetServingNetwork . . . 847
9.8.3.58 unpack_nas_GetServingNetworkCapabilities . . . 847
9.8.3.59 unpack_nas_GetSignalStrengths . . . 847
9.8.3.60 unpack_nas_InitiateDomainAttach . . . 848
9.8.3.61 unpack_nas_PerformNetworkScan . . . 848
9.8.3.62 unpack_nas_SetACCOLC . . . 848
9.8.3.63 unpack_nas_SetCDMANetworkParameters . . . 849
9.8.3.64 unpack_nas_SetDataCapabilitiesCallback_ind . . . 849
9.8.3.65 unpack_nas_SetEventReportInd . . . 849
9.8.3.66 unpack_nas_SetLURejectCallback . . . 849
9.8.3.67 unpack_nas_SetNasLTECphyCaIndCallback_ind . . . 850
9.8.3.68 unpack_nas_SetNetworkPreference . . . 850
9.8.3.69 unpack_nas_SetRFInfoCallback . . . 850
9.8.3.70 unpack_nas_SetRoamingIndicatorCallback_ind . . . 851
9.8.3.71 unpack_nas_SetServingSystemCallback_ind . . . 851
9.8.3.72 unpack_nas_SLQSConfigSigInfo . . . 851
9.8.3.73 unpack_nas_SLQSGetErrorRate . . . 852
9.8.3.74 unpack_nas_SlqsGetLTECphyCAInfo . . . 852
9.8.3.75 unpack_nas_SLQSGetNetworkTime . . . 852
9.8.3.76 unpack_nas_SLQSGetOperatorNameData . . . 853
9.8.3.77 unpack_nas_SLQSGetPLMNName . . . 853
9.8.3.78 unpack_nas_SLQSGetServingSystem . . . 853
9.8.3.79 unpack_nas_SLQSGetSignalStrength . . . 854
9.8.3.80 unpack_nas_SLQSGetSysInfo . . . 854
9.8.3.81 unpack_nas_SLQSGetSysSelectionPref . . . 855
9.8.3.82 unpack_nas_SLQSInitiateNetworkRegistration . . . 855
9.8.3.83 unpack_nas_SLQSNasConfigSigInfo2 . . . 855
9.8.3.84 unpack_nas_SLQSNasGet3GPP2Subscription . . . 855
9.8.3.85 unpack_nas_SLQSNasGetCellLocationInfo . . . 856
9.8.3.86 unpack_nas_SLQSNasGetHDRColorCode . . . 856
9.8.3.87 unpack_nas_SLQSNasGetSigInfo . . . 856
9.8.3.88 unpack_nas_SLQSNasGetTxRxInfo . . . 857
9.8.3.89 unpack_nas_SLQSNasIndicationRegisterExt . . . 857
9.8.3.90 unpack_nas_SLQSNasNetworkTimeCallBack_ind . . . 858
9.8.3.91 unpack_nas_SLQSNasSigInfoCallback_ind . . . 858

9.8.3.92 unpack_nas_SLQSNASSwiGetChannelLock . . . 858
9.8.3.93 unpack_nas_SLQSNasSwiIndicationRegister . . . 858
9.8.3.94 unpack_nas_SLQSNasSwiModemStatus . . . 859
9.8.3.95 unpack_nas_SLQSNasSwiOTAMessageCallback_ind . . . 859
9.8.3.96 unpack_nas_SLQSNASSwiSetChannelLock . . . 859
9.8.3.97 unpack_nas_SLQSNasSysInfoCallback_ind . . . 860
9.8.3.98 unpack_nas_SLQSNasTimerCallback_ind . . . 860
9.8.3.99 unpack_nas_SLQSSetBandPreference . . . 860
9.8.3.100 unpack_nas_SLQSSetSignalStrengthsCallback . . . 861
9.8.3.101 unpack_nas_SLQSSetSysSelectionPref . . . 861
9.8.3.102 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind . . . 861
9.8.3.103 unpack_nas_SLQSSwiGetHDRPersonality . . . 861
9.8.3.104 unpack_nas_SLQSSwiGetHDRProtSubtype . . . 862
9.8.3.105 unpack_nas_SLQSSwiGetHRPDStats . . . 862
9.8.3.106 unpack_nas_SLQSSwiGetLteCQI . . . 862
9.8.3.107 unpack_nas_SLQSSwiGetLteSccRxInfo . . . 863
9.8.3.108 unpack_nas_SLQSSwiNetworkDebug . . . 863
9.8.3.109 unpack_nas_SLQSSwiPSDetach . . . 863
9.9 pds.h File Reference . . . 864
9.9.1 Function Documentation . . . 866
9.9.1.1 pack_pds_ForceXTRADownload . . . 866
9.9.1.2 pack_pds_GetPDSDefaults . . . 866
9.9.1.3 pack_pds_GetPDSState . . . 867
9.9.1.4 pack_pds_GetPortAutomaticTracking . . . 867
9.9.1.5 pack_pds_GetServiceAutomaticTracking . . . 867
9.9.1.6 pack_pds_GetXTRAAutomaticDownload . . . 868
9.9.1.7 pack_pds_GetXTRANetwork . . . 868
9.9.1.8 pack_pds_GetXTRAValidity . . . 868
9.9.1.9 pack_pds_PDSInjectTimeReference . . . 869
9.9.1.10 pack_pds_ResetPDSData . . . 869
9.9.1.11 pack_pds_SetEventReportCallback . . . 869
9.9.1.12 pack_pds_SetPDSDefaults . . . 870
9.9.1.13 pack_pds_SetPDSState . . . 870
9.9.1.14 pack_pds_SetPortAutomaticTracking . . . 870
9.9.1.15 pack_pds_SetServiceAutomaticTracking . . . 871
9.9.1.16 pack_pds_SetXTRAAutomaticDownload . . . 871
9.9.1.17 pack_pds_SetXTRANetwork . . . 872
9.9.1.18 pack_pds_SLQSGetAGPSConfig . . . 872
9.9.1.19 pack_pds_SLQSGetGPSStateInfo . . . 872
9.9.1.20 pack_pds_SLQSPDSDeterminePosition . . . 873

9.9.1.21 pack_pds_SLQSPDSInjectAbsoluteTimeReference . . . 873
9.9.1.22 pack_pds_SLQSPDSInjectPositionData . . . 873
9.9.1.23 pack_pds_SLQSSetAGPSConfig . . . 874
9.9.1.24 pack_pds_SLQSSetPositionMethodState . . . 874
9.9.1.25 pack_pds_StartPDSTrackingSessionExt . . . 874
9.9.1.26 pack_pds_StopPDSTrackingSession . . . 875
9.9.1.27 unpack_pds_ForceXTRADownload . . . 875
9.9.1.28 unpack_pds_GetPDSDefaults . . . 875
9.9.1.29 unpack_pds_GetPDSState . . . 876
9.9.1.30 unpack_pds_GetPortAutomaticTracking . . . 876
9.9.1.31 unpack_pds_GetServiceAutomaticTracking . . . 876
9.9.1.32 unpack_pds_GetXTRAAutomaticDownload . . . 877
9.9.1.33 unpack_pds_GetXTRANetwork . . . 877
9.9.1.34 unpack_pds_GetXTRAValidity . . . 878
9.9.1.35 unpack_pds_PDSInjectTimeReference . . . 878
9.9.1.36 unpack_pds_ResetPDSData . . . 878
9.9.1.37 unpack_pds_SetEventReport_Ind . . . 879
9.9.1.38 unpack_pds_SetEventReportCallback . . . 879
9.9.1.39 unpack_pds_SetPDSDefaults . . . 879
9.9.1.40 unpack_pds_SetPDSState . . . 880
9.9.1.41 unpack_pds_SetPdsState_Ind . . . 880
9.9.1.42 unpack_pds_SetPortAutomaticTracking . . . 880
9.9.1.43 unpack_pds_SetServiceAutomaticTracking . . . 881
9.9.1.44 unpack_pds_SetXTRAAutomaticDownload . . . 881
9.9.1.45 unpack_pds_SetXTRANetwork . . . 881
9.9.1.46 unpack_pds_SLQSGetAGPSConfig . . . 881
9.9.1.47 unpack_pds_SLQSGetGPSStateInfo . . . 882
9.9.1.48 unpack_pds_SLQSPDSDeterminePosition . . . 882
9.9.1.49 unpack_pds_SLQSPDSInjectAbsoluteTimeReference . . . 882
9.9.1.50 unpack_pds_SLQSPDSInjectPositionData . . . 883
9.9.1.51 unpack_pds_SLQSSetAGPSConfig . . . 883
9.9.1.52 unpack_pds_SLQSSetPositionMethodState . . . 883
9.9.1.53 unpack_pds_StartPDSTrackingSessionExt . . . 884
9.9.1.54 unpack_pds_StopPDSTrackingSession . . . 884
9.10 qaGobiApiTableBandClasses.h File Reference . . . 884
9.10.1 Detailed Description . . . 884
9.10.2 Band Classes (Value - Description) . . . 884
9.10.2.1 LTE Bands . . . 886
9.11 qaGobiApiTableCallControlReturnReasons.h File Reference . . . 887
9.11.1 Detailed Description . . . 887

9.11.2 Call Control Result Reasons (Value - Name - Description) . . . 887
9.12 qaGobiApiTableCallEndReasons.h File Reference . . . 888
9.12.1 Detailed Description . . . 888
9.12.2 Call end reason codes (Code - Reason) . . . 888
9.12.2.1 Technology-agnostic call end reasons . . . 888
9.12.2.2 EVDO CDMA 1xEV-DO . . . 889
9.12.2.3 WCDMA/GSM call end reasons . . . 889
9.12.2.4 EVDO CDMA 1xEV-DO . . . 892
9.12.2.5 call end reason type . . . 892
9.12.2.6 Mobile IP call end reasons (Type=1) . . . 892
9.12.2.7 Internal call end reasons (Type=2) . . . 894
9.12.2.8 Call Manager defined call end reasons (Type=3) . . . 896
9.12.2.9 3GPP specification defined call end reasons (Type=6) . . . 901
9.12.2.10 PPP call end reasons (Type=7) . . . 903
9.12.2.11 EHRPD call end reasons (Type=8) . . . 903
9.12.2.12 IPV6 call end reasons (Type=9) . . . 904
9.13 qaGobiApiTableCarrierCodes.h File Reference . . . 904
9.13.1 Detailed Description . . . 905
9.13.2 Carrier Codes (Number - Carrier) . . . 905
9.14 qaGobiApiTableCodingScheme.h File Reference . . . 906
9.14.1 Detailed Description . . . 906
9.14.2 Call Control Result Reasons (Value - Name - Description) . . . 907
9.14.2.1 Use of bits 3..0\n\n . . . 907
9.14.3 Coding Group Bits 7..4(0001) . . . 907
9.14.3.1 use of bits 3..0 . . . 907
9.14.4 Coding Group Bits 7..4(0010) . . . 907
9.14.4.1 use of bits 3..0 . . . 907
9.14.5 Coding Group Bits 7..4(0011) . . . 908
9.14.5.1 use of bits 3..0 . . . 908
9.14.6 Coding Group Bits 7..4(01xx) . . . 908
9.14.6.1 use of bits 3..0 . . . 908
9.14.7 Coding Group Bits 7..4(1001) . . . 908
9.14.7.1 Reserved coding groups . . . 908
9.14.8 Coding Group Bits 7..4(1010..1101) . . . 909
9.14.8.1 Reserved coding groups . . . 909
9.14.9 Coding Group Bits 7..4(1110) . . . 909
9.14.9.1 Defined by the WAP Forum . . . 909
9.14.10 Coding Group Bits 7..4 (1111) . . . 909
9.14.10.1 Data coding / message handling . . . 909
9.14.11 Macro Definition Documentation . . . 909

9.14.11.1 __GOBI_API_CODING_SCHEME_H__ . . . 909
9.15 qaGobiApiTableGpsCapabilityCodes.h File Reference . . . 909
9.15.1 Detailed Description . . . 909
9.15.2 GPS capability (Value - Capability) . . . 909
9.16 qaGobiApiTablePowerModes.h File Reference . . . 910
9.16.1 Detailed Description . . . 910
9.16.2 Power Modes (Value - Description) . . . 910
9.17 qaGobiApiTableRadioInterfaces.h File Reference . . . 910
9.17.1 Detailed Description . . . 910
9.17.2 Radio interface . . . 910
9.17.2.1 Technology (Value - Radio Interface Technology) . . . 910
9.18 qaGobiApiTableRegionCodes.h File Reference . . . 911
9.18.1 Detailed Description . . . 911
9.18.2 Region Codes (Code - Region) . . . 911
9.19 qaGobiApiTableServiceOptions.h File Reference . . . 911
9.19.1 Detailed Description . . . 911
9.19.2 Service Option codes (Code - Reason) . . . 912
9.19.2.1 Description . . . 912
9.20 qaGobiApiTableSupServiceInfoClasses.h File Reference . . . 914
9.20.1 Detailed Description . . . 914
9.20.2 Supplementary Service Information Classes (Value - Service Class) . . . 914
9.21 qaGobiApiTableSwiAudio.h File Reference . . . 914
9.21.1 Detailed Description . . . 914
9.21.2 ACDB Device (Device ID - description) . . . 914
9.21.3 Physical Interface (Device ID - description - Interface parameters) . . . 915
9.22 qaGobiApiTableSwiOMADMSessionStatus.h File Reference . . . 915
9.22.1 Detailed Description . . . 915
9.22.2 OMA DM Session Status (Session Status - Meaning - Usage) . . . 915
9.23 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference . . . 916
9.23.1 Detailed Description . . . 916
9.23.2 OMA DM Update Complete Status (Update Complete Status - Meaning - Usage) . . . 916
9.24 qaGobiApiTableVoiceCallEndReasons.h File Reference . . . 917
9.24.1 Detailed Description . . . 917
9.24.2 Voice Call and supplementary services end reason codes (Code - Reason) . . . 917
9.24.2.1 General . . . 917
9.24.2.2 service Errors . . . 919
9.24.2.3 control cause values . . . 920
9.24.2.4 reject causes . . . 922
9.24.2.5 reject causes . . . 922
9.24.2.6 reject causes . . . 922

9.24.2.7 stratum reject causes . . . 923
9.24.2.8 reject causes . . . 923
9.24.2.9 IP end reasons . . . 923
9.25 qmerrno.h File Reference . . . 924
9.25.1 Enumeration Type Documentation . . . 926
9.25.1.1 eQCWWANError . . . 926
9.25.1.2 qm_wds_ds_profile_extended_err_codes . . . 931
9.26 qos.h File Reference . . . 931
9.26.1 Macro Definition Documentation . . . 933
9.26.1.1 LITEQMI_MAX_QOS_FILTERS . . . 933
9.26.1.2 LITEQMI_MAX_QOS_FLOW_PER_APN_STATS . . . 933
9.26.1.3 LITEQMI_MAX_QOS_FLOWS . . . 933
9.26.2 Function Documentation . . . 933
9.26.2.1 pack_qos_SLQSQosGetNetworkStatus . . . 933
9.26.2.2 pack_qos_SLQSQosSwiReadApnExtraParams . . . 933
9.26.2.3 pack_qos_SLQSQosSwiReadDataStats . . . 934
9.26.2.4 pack_qos_SLQSSetQosEventCallback . . . 935
9.26.2.5 unpack_qos_SLQSQosGetNetworkStatus . . . 935
9.26.2.6 unpack_qos_SLQSQosSwiReadApnExtraParams . . . 936
9.26.2.7 unpack_qos_SLQSQosSwiReadDataStats . . . 936
9.26.2.8 unpack_qos_SLQSSetQosEventCallback . . . 937
9.26.2.9 unpack_qos_SLQSSetQosEventCallback_ind . . . 937
9.26.2.10 unpack_qos_SLQSSetQosNWStatusCallback_ind . . . 938
9.26.2.11 unpack_qos_SLQSSetQosPriEventCallback_ind . . . 938
9.26.2.12 unpack_qos_SLQSSetQosStatusCallback_ind . . . 939
9.27 rms.h File Reference . . . 939
9.27.1 Function Documentation . . . 940
9.27.1.1 pack_rms_GetSMSWake . . . 940
9.27.1.2 pack_rms_SetSMSWake . . . 940
9.27.1.3 unpack_rms_GetSMSWake . . . 940
9.27.1.4 unpack_rms_SetSMSWake . . . 941
9.28 sar.h File Reference . . . 941
9.28.1 Function Documentation . . . 941
9.28.1.1 pack_sar_SLQSGetRfSarState . . . 941
9.28.1.2 pack_sar_SLQSSetRfSarState . . . 942
9.28.1.3 unpack_sar_SLQSGetRfSarState . . . 942
9.28.1.4 unpack_sar_SLQSSetRfSarState . . . 943
9.29 sms.h File Reference . . . 943
9.29.1 Macro Definition Documentation . . . 946
9.29.1.1 MAX_CDMA_ENC_MO_TXT_MSG_SIZE . . . 946

9.29.1.2 MAX_MS_TRANSFER_ROUTE_MSG . . . 946
9.29.1.3 MAX_MSC_ADDRESS_SIZE . . . 946
9.29.1.4 MAX_MSE_TWS_MSG . . . 946
9.29.1.5 MAX_SMS_LIST_SIZE . . . 946
9.29.1.6 MAX_SMS_MESSAGE_SIZE . . . 946
9.29.1.7 SMS_CONFIG_LEN . . . 946
9.29.1.8 SMS_MAX_SMS_ROUTES . . . 946
9.29.1.9 SMS_NUM_OF_SET . . . 946
9.29.1.10 SMSC_TYPE_LEN . . . 946
9.29.2 Typedef Documentation . . . 947
9.29.2.1 sMSCAddressInfo . . . 947
9.29.2.2 sMSEtwsMessageInfo . . . 947
9.29.2.3 sMSEtwsPlmnInfo . . . 947
9.29.2.4 sMSMessageModeInfo . . . 947
9.29.2.5 sMSMTMessageInfo . . . 947
9.29.2.6 sMSOnIMSInfo . . . 947
9.29.2.7 sMSTransferRouteMTMessageInfo . . . 948
9.29.3 Enumeration Type Documentation . . . 948
9.29.3.1 eqmiCbkSetStatus . . . 948
9.29.4 Function Documentation . . . 948
9.29.4.1 pack_sms_GetSMSCAddress . . . 948
9.29.4.2 pack_sms_SaveSMS . . . 949
9.29.4.3 pack_sms_SendSMS . . . 949
9.29.4.4 pack_sms_SetNewSMSCallback . . . 949
9.29.4.5 pack_sms_SetSMSCAddress . . . 950
9.29.4.6 pack_sms_SLQSDeleteSMS . . . 950
9.29.4.7 pack_sms_SLQSGetIndicationRegister . . . 950
9.29.4.8 pack_sms_SLQSGetMessageWaiting . . . 951
9.29.4.9 pack_sms_SLQSGetSMS . . . 951
9.29.4.10 pack_sms_SLQSGetSmsBroadcastConfig . . . 952
9.29.4.11 pack_sms_SLQSGetSMSList . . . 952
9.29.4.12 pack_sms_SLQSGetTransLayerInfo . . . 952
9.29.4.13 pack_sms_SLQSGetTransNWRegInfo . . . 953
9.29.4.14 pack_sms_SLQSModifySMSStatus . . . 953
9.29.4.15 pack_sms_SLQSSendAsyncSMS . . . 953
9.29.4.16 pack_sms_SLQSSetIndicationRegister . . . 954
9.29.4.17 pack_sms_SLQSSetSmsBroadcastActivation . . . 954
9.29.4.18 pack_sms_SLQSSetSmsBroadcastConfig . . . 954
9.29.4.19 pack_sms_SLQSSetSmsStorage . . . 955
9.29.4.20 pack_sms_SLQSSmsGetMaxStorageSize . . . 955

9.29.4.21 pack_sms_SLQSSmsGetMessageProtocol . . . 956
9.29.4.22 pack_sms_SLQSSmsSetRoutes . . . 956
9.29.4.23 pack_sms_SLQSSwiGetSMSStorage . . . 956
9.29.4.24 unpack_sms_GetSMSCAddress . . . 957
9.29.4.25 unpack_sms_SaveSMS . . . 957
9.29.4.26 unpack_sms_SendSMS . . . 957
9.29.4.27 unpack_sms_SetNewSMSCallback . . . 958
9.29.4.28 unpack_sms_SetNewSMSCallback_ind . . . 958
9.29.4.29 unpack_sms_SetSMSCAddress . . . 958
9.29.4.30 unpack_sms_SLQSDeleteSMS . . . 959
9.29.4.31 unpack_sms_SLQSGetIndicationRegister . . . 959
9.29.4.32 unpack_sms_SLQSGetMessageWaiting . . . 959
9.29.4.33 unpack_sms_SLQSGetSMS . . . 960
9.29.4.34 unpack_sms_SLQSGetSmsBroadcastConfig . . . 960
9.29.4.35 unpack_sms_SLQSGetSMSList . . . 960
9.29.4.36 unpack_sms_SLQSGetTransLayerInfo . . . 961
9.29.4.37 unpack_sms_SLQSGetTransNWRegInfo . . . 961
9.29.4.38 unpack_sms_SLQSModifySMSStatus . . . 961
9.29.4.39 unpack_sms_SLQSSendAsyncSMS . . . 962
9.29.4.40 unpack_sms_SLQSSetIndicationRegister . . . 962
9.29.4.41 unpack_sms_SLQSSetSmsBroadcastActivation . . . 962
9.29.4.42 unpack_sms_SLQSSetSmsBroadcastConfig . . . 963
9.29.4.43 unpack_sms_SLQSSetSmsStorage . . . 963
9.29.4.44 unpack_sms_SLQSSmsGetMaxStorageSize . . . 963
9.29.4.45 unpack_sms_SLQSSmsGetMessageProtocol . . . 964
9.29.4.46 unpack_sms_SLQSSmsSetRoutes . . . 964
9.29.4.47 unpack_sms_SLQSSwiGetSMSStorage . . . 964
9.29.4.48 unpack_sms_SLQSWmsMemoryFullCallBack_ind . . . 965
9.30 swiavms.h File Reference . . . 965
9.30.1 Macro Definition Documentation . . . 967
9.30.1.1 LITEQMI_MAX_SWIOMA_STR_LEN . . . 967
9.30.1.2 MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER . . . 967
9.30.1.3 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH . . . 967
9.30.1.4 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH . . . 967
9.30.1.5 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH . . . 967
9.30.1.6 MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH . . . 967
9.30.1.7 MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH . . . 967

9.30.1.8 MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS . 968
9.30.1.9 MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 968
9.30.1.10 MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 968
9.30.2 Typedef Documentation 968
9.30.2.1 PackSwiAvmsSetSettingsAPNInfo 968
9.30.2.2 PackSwiAvmsSetSettingsConnectionRetryTimers 968
9.30.2.3 PackSwiAvmsSetSettingsPeriodInfo 968
9.30.2.4 PackSwiAVMSSettingsAPNInfo 968
9.30.2.5 PackSwiAVMSSettingsConnectionRetryTimers 968
9.30.2.6 PackSwiAVMSSettingsPeriodsInfo 968
9.30.2.7 unpack_swiavms_SLQSAVMSEventReportInd_t 968
9.30.2.8 unpack_swiavms_SLQSAVMSSessionGetInfo_t 969
9.30.2.9 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo 969
9.30.2.10 UnpackSwiAvmsEventReportConfig 970
9.30.2.11 UnpackSwiAvmsEventReportConnectionRequest 970
9.30.2.12 UnpackSwiAvmsEventReportDataSessionStatus 971
9.30.2.13 UnpackSwiAvmsEventReportHTTPStatus 971
9.30.2.14 UnpackSwiAvmsEventReportNotification 972
9.30.2.15 UnpackSwiAvmsEventReportPackageID 973
9.30.2.16 UnpackSwiAvmsEventReportRegStatus 973
9.30.2.17 UnpackSwiAvmsEventReportSessionType 974
9.30.2.18 UnpackSwiAvmsEventReportWAMSParamChange 974
9.30.3 Function Documentation 974
9.30.3.1 pack_swiavms_SLQSAVMSGetSettings 974
9.30.3.2 pack_swiavms_SLQSAVMSSendSelection 975
9.30.3.3 pack_swiavms_SLQSAVMSSessionGetInfo 975
9.30.3.4 pack_swiavms_SLQSAvmsSetEventReport 976
9.30.3.5 pack_swiavms_SLQSAVMSSetSettings 976
9.30.3.6 pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField 977
9.30.3.7 pack_swiavms_SLQSAVMSStartSession 978
9.30.3.8 pack_swiavms_SLQSAVMSStopSession 978
9.30.3.9 unpack_swiavms_SLQSAVMSEventReportInd 979
9.30.3.10 unpack_swiavms_SLQSAVMSGetSettings 980
9.30.3.11 unpack_swiavms_SLQSAVMSSendSelection 980
9.30.3.12 unpack_swiavms_SLQSAVMSSessionGetInfo 980
9.30.3.13 unpack_swiavms_SLQSAvmsSetEventReport 981
9.30.3.14 unpack_swiavms_SLQSAVMSSetSettings 981
9.30.3.15 unpack_swiavms_SLQSAVMSStartSession 982
9.30.3.16 unpack_swiavms_SLQSAVMSStopSession 982
9.31 SwiDataTypes.h File Reference 983

9.31.1 Detailed Description . . . 983
9.31.2 Macro Definition Documentation . . . 983
9.31.2.1 QMI_NO_LTE_FW_SUPPORT . . . 983
9.31.2.2 QMI_TLV_PLACEHOLDER . . . 983
9.31.2.3 SWI_API . . . 983
9.31.2.4 UNUSEDPARAM . . . 984
9.31.3 Typedef Documentation . . . 984
9.31.3.1 BOOL . . . 984
9.31.3.2 BYTE . . . 984
9.31.3.3 CHAR . . . 984
9.31.3.4 FLOAT . . . 984
9.31.3.5 INT32 . . . 984
9.31.3.6 INT8 . . . 984
9.31.3.7 LPCSTR . . . 984
9.31.3.8 SHORT . . . 984
9.31.3.9 ULONG . . . 984
9.31.3.10 ULONGLONG . . . 984
9.31.3.11 USHORT . . . 984
9.31.3.12 WORD . . . 984
9.32 swiloc.h File Reference . . . 984
9.32.1 Function Documentation . . . 984
9.32.1.1 pack_swiloc_SwiLocGetAutoStart . . . 984
9.32.1.2 pack_swiloc_SwiLocSetAutoStart . . . 985
9.32.1.3 unpack_swiloc_SwiLocGetAutoStart . . . 985
9.32.1.4 unpack_swiloc_SwiLocSetAutoStart . . . 986
9.33 swioma.h File Reference . . . 986
9.33.1 Macro Definition Documentation . . . 987
9.33.1.1 LITEQMI_MAX_SWIOMA_STR_LEN . . . 987
9.33.2 Function Documentation . . . 987
9.33.2.1 pack_swioma_SLQSOMADMAlertCallback . . . 987
9.33.2.2 pack_swioma_SLQSOMADMCancelSession . . . 987
9.33.2.3 pack_swioma_SLQSOMADMGetSessionInfo . . . 988
9.33.2.4 pack_swioma_SLQSOMADMGetSettings . . . 989
9.33.2.5 pack_swioma_SLQSOMADMSendSelection . . . 989
9.33.2.6 pack_swioma_SLQSOMADMSetSettings . . . 990
9.33.2.7 pack_swioma_SLQSOMADMStartSession . . . 991
9.33.2.8 unpack_swioma_SLQSOMADMAlertCallback . . . 991
9.33.2.9 unpack_swioma_SLQSOMADMAlertCallback_ind . . . 992
9.33.2.10 unpack_swioma_SLQSOMADMCancelSession . . . 992
9.33.2.11 unpack_swioma_SLQSOMADMGetSessionInfo . . . 993

9.33.2.12 unpack_swioma_SLQSOMADMGetSettings . . . 993
9.33.2.13 unpack_swioma_SLQSOMADMSendSelection . . . 994
9.33.2.14 unpack_swioma_SLQSOMADMSetSettings . . . 994
9.33.2.15 unpack_swioma_SLQSOMADMStartSession . . . 994
9.34 SWIWWANCMAPI.h File Reference . . . 995
9.35 uim.h File Reference . . . 995
9.35.1 Macro Definition Documentation . . . 998
9.35.1.1 MAX_DESCRIPTION_LENGTH . . . 998
9.35.1.2 MAX_ICCID_LENGTH . . . 998
9.35.1.3 MAX_NO_OF_APPLICATIONS . . . 998
9.35.1.4 MAX_NO_OF_SLOTS . . . 998
9.35.1.5 MAX_SLOTS_STATUS . . . 998
9.35.1.6 UIM_MAX_ACTIVE_PERS_FEATURES . . . 998
9.35.1.7 UIM_MAX_CONTENT_LENGTH . . . 998
9.35.1.8 UIM_MAX_DESCRIPTION_LENGTH . . . 998
9.35.1.9 UIM_MAX_NO_OF_APPLICATIONS . . . 998
9.35.1.10 UIM_MAX_NO_OF_SLOTS . . . 998
9.35.1.11 UIM_UINT8_MAX_STRING_SZ . . . 998
9.35.2 Function Documentation . . . 998
9.35.2.1 pack_uim_ChangePin . . . 998
9.35.2.2 pack_uim_GetCardStatus . . . 998
9.35.2.3 pack_uim_ReadTransparent . . . 999
9.35.2.4 pack_uim_SetPinProtection . . . 999
9.35.2.5 pack_uim_SLQSUIMAuthenticate . . . 999
9.35.2.6 pack_uim_SLQSUIMDepersonalization . . . 1000
9.35.2.7 pack_uim_SLQSUIMEventRegister . . . 1000
9.35.2.8 pack_uim_SLQSUIMGetConfiguration . . . 1001
9.35.2.9 pack_uim_SLQSUIMGetFileAttributes . . . 1001
9.35.2.10 pack_uim_SLQSUIMGetSlotsStatus . . . 1001
9.35.2.11 pack_uim_SLQSUIMPowerDown . . . 1002
9.35.2.12 pack_uim_SLQSUIMPowerUp . . . 1002
9.35.2.13 pack_uim_SLQSUIMRefreshComplete . . . 1002
9.35.2.14 pack_uim_SLQSUIMRefreshGetLastEvent . . . 1003
9.35.2.15 pack_uim_SLQSUIMRefreshOK . . . 1003
9.35.2.16 pack_uim_SLQSUIMRefreshRegister . . . 1003
9.35.2.17 pack_uim_SLQSUIMReset . . . 1004
9.35.2.18 pack_uim_SLQSUIMSwitchSlot . . . 1004
9.35.2.19 pack_uim_UnblockPin . . . 1005
9.35.2.20 pack_uim_VerifyPin . . . 1005
9.35.2.21 unpack_uim_ChangePin . . . 1005

9.35.2.22 unpack_uim_GetCardStatus . . . 1006
9.35.2.23 unpack_uim_ReadTransparent . . . 1006
9.35.2.24 unpack_uim_SetPinProtection . . . 1006
9.35.2.25 unpack_uim_SetUimSlotStatusChangeCallback_ind . . . 1007
9.35.2.26 unpack_uim_SLQSUIMAuthenticate . . . 1007
9.35.2.27 unpack_uim_SLQSUIMDepersonalization . . . 1007
9.35.2.28 unpack_uim_SLQSUIMEventRegister . . . 1008
9.35.2.29 unpack_uim_SLQSUIMGetConfiguration . . . 1008
9.35.2.30 unpack_uim_SLQSUIMGetFileAttributes . . . 1009
9.35.2.31 unpack_uim_SLQSUIMGetSlotsStatus . . . 1009
9.35.2.32 unpack_uim_SLQSUIMPowerDown . . . 1009
9.35.2.33 unpack_uim_SLQSUIMPowerUp . . . 1010
9.35.2.34 unpack_uim_SLQSUIMRefreshComplete . . . 1010
9.35.2.35 unpack_uim_SLQSUIMRefreshGetLastEvent . . . 1010
9.35.2.36 unpack_uim_SLQSUIMRefreshOK . . . 1011
9.35.2.37 unpack_uim_SLQSUIMRefreshRegister . . . 1011
9.35.2.38 unpack_uim_SLQSUIMReset . . . 1011
9.35.2.39 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind . . . 1011
9.35.2.40 unpack_uim_SLQSUIMSwitchSlot . . . 1012
9.35.2.41 unpack_uim_UnblockPin . . . 1012
9.35.2.42 unpack_uim_VerifyPin . . . 1013
9.36 voice.h File Reference . . . 1013
9.36.1 Macro Definition Documentation . . . 1017
9.36.1.1 BARRING_PASSWORD_LENGTH . . . 1017
9.36.1.2 MAX_VOICE_CALL_NO_LEN . . . 1017
9.36.1.3 MAX_VOICE_DESCRIPTION_LENGTH . . . 1017
9.36.1.4 MAXVOICEUSSDLENGTH . . . 1017
9.36.1.5 VOICE_MAX_NO_OF_CALLS . . . 1017
9.36.2 Enumeration Type Documentation . . . 1017
9.36.2.1 liteServiceClassInformation . . . 1017
9.36.3 Function Documentation . . . 1018
9.36.3.1 pack_voice_AnswerUSSD . . . 1018
9.36.3.2 pack_voice_CancelUSSD . . . 1018
9.36.3.3 pack_voice_OriginateUSSD . . . 1018
9.36.3.4 pack_voice_SLQSOriginateUSSD . . . 1019
9.36.3.5 pack_voice_SLQSVoiceALSSelectLine . . . 1019
9.36.3.6 pack_voice_SLQSVoiceALSSetLineSwitching . . . 1020
9.36.3.7 pack_voice_SLQSVoiceAnswerCall . . . 1020
9.36.3.8 pack_voice_SLQSVoiceBindSubscription . . . 1020
9.36.3.9 pack_voice_SLQSVoiceBurstDTMF . . . 1021

9.36.3.10 pack_voice_SLQSVoiceDialCall . . . 1021
9.36.3.11 pack_voice_SLQSVoiceEndCall . . . 1021
9.36.3.12 pack_voice_SLQSVoiceGetAllCallInfo . . . 1022
9.36.3.13 pack_voice_SLQSVoiceGetCallBarring . . . 1022
9.36.3.14 pack_voice_SLQSVoiceGetCallForwardingStatus . . . 1023
9.36.3.15 pack_voice_SLQSVoiceGetCallInfo . . . 1023
9.36.3.16 pack_voice_SLQSVoiceGetCallWaiting . . . 1023
9.36.3.17 pack_voice_SLQSVoiceGetCLIP . . . 1024
9.36.3.18 pack_voice_SLQSVoiceGetCLIR . . . 1024
9.36.3.19 pack_voice_SLQSVoiceGetCNAP . . . 1024
9.36.3.20 pack_voice_SLQSVoiceGetCOLP . . . 1025
9.36.3.21 pack_voice_SLQSVoiceGetCOLR . . . 1025
9.36.3.22 pack_voice_SLQSVoiceGetConfig . . . 1025
9.36.3.23 pack_voice_SLQSVoiceIndicationRegister . . . 1026
9.36.3.24 pack_voice_SLQSVoiceManageCalls . . . 1026
9.36.3.25 pack_voice_SLQSVoiceOrigUSSDNoWait . . . 1026
9.36.3.26 pack_voice_SLQSVoiceSendFlash . . . 1027
9.36.3.27 pack_voice_SLQSVoiceSetCallBarringPassword . . . 1027
9.36.3.28 pack_voice_SLQSVoiceSetConfig . . . 1028
9.36.3.29 pack_voice_SLQSVoiceSetPreferredPrivacy . . . 1028
9.36.3.30 pack_voice_SLQSVoiceSetSUPSService . . . 1028
9.36.3.31 pack_voice_SLQSVoiceStartContDTMF . . . 1029
9.36.3.32 pack_voice_SLQSVoiceStopContDTMF . . . 1029
9.36.3.33 unpack_voice_AnswerUSSD . . . 1029
9.36.3.34 unpack_voice_CancelUSSD . . . 1030
9.36.3.35 unpack_voice_OriginateUSSD . . . 1030
9.36.3.36 unpack_voice_SLQSOriginateUSSD . . . 1030
9.36.3.37 unpack_voice_SLQSVoiceALSSelectLine . . . 1031
9.36.3.38 unpack_voice_SLQSVoiceALSSetLineSwitching . . . 1031
9.36.3.39 unpack_voice_SLQSVoiceAnswerCall . . . 1031
9.36.3.40 unpack_voice_SLQSVoiceBindSubscription . . . 1032
9.36.3.41 unpack_voice_SLQSVoiceBurstDTMF . . . 1032
9.36.3.42 unpack_voice_SLQSVoiceDialCall . . . 1032
9.36.3.43 unpack_voice_SLQSVoiceEndCall . . . 1033
9.36.3.44 unpack_voice_SLQSVoiceGetAllCallInfo . . . 1033
9.36.3.45 unpack_voice_SLQSVoiceGetCallBarring . . . 1033
9.36.3.46 unpack_voice_SLQSVoiceGetCallForwardingStatus . . . 1034
9.36.3.47 unpack_voice_SLQSVoiceGetCallInfo . . . 1034
9.36.3.48 unpack_voice_SLQSVoiceGetCallWaiting . . . 1034
9.36.3.49 unpack_voice_SLQSVoiceGetCLIP . . . 1035

9.36.3.50 unpack_voice_SLQSVoiceGetCLIR . . . 1035
9.36.3.51 unpack_voice_SLQSVoiceGetCNAP . . . 1035
9.36.3.52 unpack_voice_SLQSVoiceGetCOLP . . . 1036
9.36.3.53 unpack_voice_SLQSVoiceGetCOLR . . . 1036
9.36.3.54 unpack_voice_SLQSVoiceGetConfig . . . 1036
9.36.3.55 unpack_voice_SLQSVoiceIndicationRegister . . . 1037
9.36.3.56 unpack_voice_SLQSVoiceManageCalls . . . 1037
9.36.3.57 unpack_voice_SLQSVoiceOrigUSSDNoWait . . . 1037
9.36.3.58 unpack_voice_SLQSVoiceSendFlash . . . 1038
9.36.3.59 unpack_voice_SLQSVoiceSetCallBarringPassword . . . 1038
9.36.3.60 unpack_voice_SLQSVoiceSetConfig . . . 1038
9.36.3.61 unpack_voice_SLQSVoiceSetPreferredPrivacy . . . 1039
9.36.3.62 unpack_voice_SLQSVoiceSetSUPSService . . . 1039
9.36.3.63 unpack_voice_SLQSVoiceStartContDTMF . . . 1039
9.36.3.64 unpack_voice_SLQSVoiceStopContDTMF . . . 1040
9.37 wds.h File Reference . . . 1040
9.37.1 Macro Definition Documentation . . . 1046
9.37.1.1 BYT_STAT_STAT_MASK . . . 1046
9.37.1.2 IPV6_ADDRESS_ARRAY_SIZE . . . 1046
9.37.1.3 MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE . . . 1046
9.37.1.4 PACK_WDS_IPV4 . . . 1046
9.37.1.5 PACK_WDS_IPV6 . . . 1046
9.37.1.6 WDS_DHCP_MAX_NUM_OPTIONS . . . 1046
9.37.1.7 WDS_DHCP_OPTION_DATA_BUF_SIZE . . . 1046
9.37.2 Typedef Documentation . . . 1046
9.37.2.1 UnpackQmiProfileInfo . . . 1046
9.37.3 Enumeration Type Documentation . . . 1046
9.37.3.1 liteQmiDataBearerMasks . . . 1046
9.37.4 Function Documentation . . . 1046
9.37.4.1 pack_wds_DHCPv4ClientLeaseChange . . . 1046
9.37.4.2 pack_wds_GetAutoconnect . . . 1047
9.37.4.3 pack_wds_GetByteTotals . . . 1047
9.37.4.4 pack_wds_GetConnectionRate . . . 1048
9.37.4.5 pack_wds_GetDataBearerTechnology . . . 1048
9.37.4.6 pack_wds_GetDefaultProfile . . . 1048
9.37.4.7 pack_wds_GetDefaultProfileNum . . . 1049
9.37.4.8 pack_wds_GetDormancyState . . . 1049
9.37.4.9 pack_wds_GetLastMobileIPError . . . 1050
9.37.4.10 pack_wds_GetMobileIP . . . 1050
9.37.4.11 pack_wds_GetMobileIPProfile . . . 1051

9.37.4.12 pack_wds_GetPacketStatistics . . . 1051
9.37.4.13 pack_wds_GetPacketStatus . . . 1051
9.37.4.14 pack_wds_GetSessionDuration . . . 1052
9.37.4.15 pack_wds_GetSessionState . . . 1052
9.37.4.16 pack_wds_RMSetTransferStatistics . . . 1053
9.37.4.17 pack_wds_SetAutoconnect . . . 1053
9.37.4.18 pack_wds_SetDefaultProfile . . . 1053
9.37.4.19 pack_wds_SetDefaultProfileNum . . . 1054
9.37.4.20 pack_wds_SetMobileIP . . . 1054
9.37.4.21 pack_wds_SetMobileIPParameters . . . 1055
9.37.4.22 pack_wds_SetMobileIPProfile . . . 1055
9.37.4.23 pack_wds_SLQSCreateProfile . . . 1055
9.37.4.24 pack_wds_SLQSDeleteProfile . . . 1056
9.37.4.25 pack_wds_SLQSGet3GPPConfigItem . . . 1056
9.37.4.26 pack_wds_SLQSGetCurrDataSystemStat . . . 1057
9.37.4.27 pack_wds_SLQSGetCurrentChannelRate . . . 1057
9.37.4.28 pack_wds_SLQSGetDataBearerTechnology . . . 1057
9.37.4.29 pack_wds_SLQSGetDUNCallInfo . . . 1058
9.37.4.30 pack_wds_SLQSGetProfileSettings . . . 1058
9.37.4.31 pack_wds_SLQSGetRuntimeSettings . . . 1059
9.37.4.32 pack_wds_SLQSModifyProfile . . . 1059
9.37.4.33 pack_wds_SLQSResetPacketStatics . . . 1060
9.37.4.34 pack_wds_SLQSSet3GPPConfigItem . . . 1060
9.37.4.35 pack_wds_SLQSSetIPFamilyPreference . . . 1060
9.37.4.36 pack_wds_SLQSSetWdsEventCallback . . . 1061
9.37.4.37 pack_wds_SLQSSGetDHCPv4ClientConfig . . . 1061
9.37.4.38 pack_wds_SLQSSGetLoopback . . . 1062
9.37.4.39 pack_wds_SLQSSSetDHCPv4ClientConfig . . . 1062
9.37.4.40 pack_wds_SLQSSSetLoopback . . . 1062
9.37.4.41 pack_wds_SLQSStartDataSession . . . 1063
9.37.4.42 pack_wds_SLQSStopDataSession . . . 1063
9.37.4.43 pack_wds_SLQSWdsGoActive . . . 1064
9.37.4.44 pack_wds_SLQSWdsGoDormant . . . 1064
9.37.4.45 pack_wds_SLQSWdsSetEventReport . . . 1065
9.37.4.46 pack_wds_SLQSWdsSwiPDPRuntimeSettings . . . 1065
9.37.4.47 unpack_wds_DHCPv4ClientLease_ind . . . 1065
9.37.4.48 unpack_wds_DHCPv4ClientLeaseChange . . . 1066
9.37.4.49 unpack_wds_GetAutoconnect . . . 1066
9.37.4.50 unpack_wds_GetByteTotals . . . 1066
9.37.4.51 unpack_wds_GetConnectionRate . . . 1067

9.37.4.52 unpack_wds_GetDataBearerTechnology . . . 1067
9.37.4.53 unpack_wds_GetDefaultProfile . . . 1067
9.37.4.54 unpack_wds_GetDefaultProfileNum . . . 1068
9.37.4.55 unpack_wds_GetDormancyState . . . 1068
9.37.4.56 unpack_wds_GetLastMobileIPError . . . 1069
9.37.4.57 unpack_wds_GetMobileIP . . . 1069
9.37.4.58 unpack_wds_GetMobileIPProfile . . . 1069
9.37.4.59 unpack_wds_GetPacketStatistics . . . 1070
9.37.4.60 unpack_wds_GetPacketStatus . . . 1070
9.37.4.61 unpack_wds_GetSessionDuration . . . 1070
9.37.4.62 unpack_wds_GetSessionState . . . 1071
9.37.4.63 unpack_wds_RMSetTransferStatistics . . . 1071
9.37.4.64 unpack_wds_RMTransferStatistics_ind . . . 1071
9.37.4.65 unpack_wds_SetAutoconnect . . . 1072
9.37.4.66 unpack_wds_SetDefaultProfile . . . 1072
9.37.4.67 unpack_wds_SetDefaultProfileNum . . . 1072
9.37.4.68 unpack_wds_SetMobileIP . . . 1073
9.37.4.69 unpack_wds_SetMobileIPParameters . . . 1073
9.37.4.70 unpack_wds_SetMobileIPProfile . . . 1073
9.37.4.71 unpack_wds_SLQSCreateProfile . . . 1074
9.37.4.72 unpack_wds_SLQSDeleteProfile . . . 1074
9.37.4.73 unpack_wds_SLQSGet3GPPConfigItem . . . 1074
9.37.4.74 unpack_wds_SLQSGetCurrDataSystemStat . . . 1075
9.37.4.75 unpack_wds_SLQSGetCurrentChannelRate . . . 1075
9.37.4.76 unpack_wds_SLQSGetDataBearerTechnology . . . 1075
9.37.4.77 unpack_wds_SLQSGetDUNCallInfo . . . 1076
9.37.4.78 unpack_wds_SLQSGetProfileSettings . . . 1076
9.37.4.79 unpack_wds_SLQSGetRuntimeSettings . . . 1076
9.37.4.80 unpack_wds_SLQSModifyProfile . . . 1077
9.37.4.81 unpack_wds_SLQSResetPacketStatics . . . 1077
9.37.4.82 unpack_wds_SLQSSet3GPPConfigItem . . . 1077
9.37.4.83 unpack_wds_SLQSSetIPFamilyPreference . . . 1078
9.37.4.84 unpack_wds_SLQSSetPacketSrvStatusCallback . . . 1078
9.37.4.85 unpack_wds_SLQSSetWdsEventCallback . . . 1078
9.37.4.86 unpack_wds_SLQSSetWdsEventCallback_ind . . . 1079
9.37.4.87 unpack_wds_SLQSSGetDHCPv4ClientConfig . . . 1079
9.37.4.88 unpack_wds_SLQSSGetLoopback . . . 1079
9.37.4.89 unpack_wds_SLQSSSetDHCPv4ClientConfig . . . 1080
9.37.4.90 unpack_wds_SLQSSSetLoopback . . . 1080
9.37.4.91 unpack_wds_SLQSStartDataSession . . . 1080

9.37.4.92 unpack_wds_SLQSStopDataSession 1081
9.37.4.93 unpack_wds_SLQSWdsGoActive 1081
9.37.4.94 unpack_wds_SLQSWdsGoDormant 1081
9.37.4.95 unpack_wds_SLQSWdsSetEventReport 1082
9.37.4.96 unpack_wds_SLQSWdsSwiPDPRuntimeSettings 1082

Index **1083**

Chapter 1

Welcome to the Sierra Wireless Linux QMI SDK API Reference Guide

This API reference guide contains information about all the modules, in the Sierra Wireless Linux QMI SDK (SLQS). Use the tabs at the top of the page to navigate the reference guide.

- Modules tab – lists all the service modules and provides a link to the API header file in each module.
- References tab – links to reference material.

1.1 Important Notice

Due to the nature of wireless communications, transmission and reception of data can never be guaranteed. Data may be delayed, corrupted (i.e., have errors) or be totally lost. Although significant delays or losses of data are rare when wireless devices such as the Sierra Wireless modem are used in a normal manner with a well-constructed network, the Sierra Wireless modem should not be used in situations where failure to transmit or receive data could result in damage of any kind to the user or any other party, including but not limited to personal injury, death, or loss of property. Sierra Wireless accepts no responsibility for damages of any kind resulting from delays or errors in data transmitted or received using the Sierra Wireless modem, or for failure of the Sierra Wireless modem to transmit or receive such data.

1.2 Limitation of Liability

The information in this manual is subject to change without notice and does not represent a commitment on the part of Sierra Wireless. SIERRA WIRELESS AND ITS AFFILIATES SPECIFICALLY DISCLAIM LIABILITY FOR ANY AND ALL DIRECT, INDIRECT, SPECIAL, GENERAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES INCLUDING, BUT NOT LIMITED TO, LOSS OF PROFITS OR REVENUE OR ANTICIPATED PROFITS OR REVENUE ARISING OUT OF THE USE OR INABILITY TO USE ANY SIERRA WIRELESS PRODUCT, EVEN IF SIERRA WIRELESS AND/OR ITS AFFILIATES HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR THEY ARE FORESEEABLE OR FOR CLAIMS BY ANY THIRD PARTY.

Notwithstanding the foregoing, in no event shall Sierra Wireless and/or its affiliates aggregate liability arising under or in connection with the Sierra Wireless product, regardless of the number of events, occurrences, or claims giving rise to liability, be in excess of the price paid by the purchaser for the Sierra Wireless product.

1.3 Patents

Patents This product may contain technology developed by or for Sierra Wireless Inc. This product includes technology licensed from QUALCOMM® 3G. This product is manufactured or sold by Sierra Wireless Inc. or its affiliates

under one or more patents licensed from InterDigital Group.

1.4 Copyright

© 2011-2015 Sierra Wireless. All rights reserved.

1.5 Trademarks

AirCard® and Heart of the Wireless Machine® are registered trademarks of Sierra Wireless. Watcher® is a trademark of Sierra Wireless, registered in the European Community. Sierra Wireless, the Sierra Wireless logo, the red wave design, and the red-tipped antenna are trademarks of Sierra Wireless. Windows® is a registered trademark of Microsoft Corporation. QUALCOMM® is a registered trademark of QUALCOMM Incorporated. Used under license. Linux is a registered trademark of Linus Torvalds. Other trademarks are the property of the respective owners.

1.6 Contact Information

If you have any questions about the Sierra Wireless Linux SDK, contact your Sierra Wireless account manager.

Consult our website for up-to-date product descriptions, documentation, application notes, firmware upgrades, troubleshooting tips, and press releases: http://www.sierrawireless.com.

Chapter 2

Module Index

2.1 Modules

Here is a list of all modules:

QMI pack/unpack (lite-qmi) 25
Streaming Download Protocol (lite-fw) 26

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Tables 27

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

_litefw_FirmwareInfo_ . . . 29
_PackSwiAvmsSetSettingsAPNInfo . . . 30
_PackSwiAvmsSetSettingsConnectionRetryTimers . . . 30
_PackSwiAvmsSetSettingsPeriodInfo . . . 31
_PackSwiAVMSSettingsAPNInfo . . . 31
_PackSwiAVMSSettingsConnectionRetryTimers . . . 31
_PackSwiAVMSSettingsPeriodsInfo . . . 32
_unpack_swiavms_SLQSAVMSEventReportInd_t . . . 32
_unpack_swiavms_SLQSAVMSSessionGetInfo_t . . . 33
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo . . . 34
_UnpackSwiAvmsEventReportConfig . . . 36
_UnpackSwiAvmsEventReportConnectionRequest . . . 37
_UnpackSwiAvmsEventReportDataSessionStatus . . . 37
_UnpackSwiAvmsEventReportHTTPStatus . . . 38
_UnpackSwiAvmsEventReportNotification . . . 40
_UnpackSwiAvmsEventReportPackageID . . . 41
_UnpackSwiAvmsEventReportRegStatus . . . 41
_UnpackSwiAvmsEventReportSessionType . . . 41
_UnpackSwiAvmsEventReportWAMSParamChange . . . 42
altSrcInfo_t . . . 42
appStats . . . 43
CarrierImage_t . . . 46
cat_AlPhaIdentifierTlv . . . 48
cat_commonEventTlv . . . 48
cat_currentCatEvent . . . 49
cat_EndProactiveSessionTlv . . . 49
cat_EventIDDataTlv . . . 50
cat_EventListTlv . . . 50
cat_RefreshTlv . . . 51
cdmaSSInfo . . . 51
connectionStatus . . . 52
crashInfoParams . . . 53
crashInformation . . . 53
currNetworkInfo . . . 54
dms_ActivationStatusTlv . . . 56
dms_OperatingModeTlv . . . 56
DMScustSettingInfo . . . 57
DMScustSettingList . . . 58

DMSgetCustomFeatureV2 . . . 59
DMSgetCustomInput . . . 59
dunchannelRate . . . 60
eriDataparams . . . 60
eTWSPLMNInfoTlv . . . 61
FMSImageElement . . . 61
FMSImageIdElement . . . 62
FMSImageIDEntries . . . 63
FMSImageList . . . 64
FMSPrefImageList . . . 65
hdrSSInfo . . . 65
image_info_t . . . 66
ipv6AddressInfo . . . 67
LibPackGPRSRequestedQoS . . . 67
LibpackProfile3GPP . . . 68
LibpackProfile3GPP2 . . . 74
LibPackprofile_3GPP . . . 79
LibPackprofile_3GPP2 . . . 85
LibPackQosClassID . . . 90
LibPackTFTIDParams . . . 91
LibPackUMTSQoS . . . 93
LibPackUMTSReqQoSSigInd . . . 95
loc_BdsSV . . . 96
loc_BdsSVInfo . . . 97
loc_CellDb . . . 97
loc_ClkInfo . . . 98
loc_GnssData . . . 99
loc_gpsTime . . . 101
loc_IPv4Info . . . 101
loc_IPv6Info . . . 102
loc_LocApplicationInfo . . . 102
loc_precisionDilution . . . 103
loc_satelliteInfo . . . 104
loc_sensorDataUsage . . . 106
loc_SV . . . 107
loc_SVInfo . . . 108
loc_svUsedforFix . . . 108
loc_urlAddr . . . 109
lteSSInfo . . . 110
messageModeTlv . . . 110
nas_acqOrderPref . . . 111
nas_ActPilotPNElement . . . 111
nas_AddCDMASysInfo . . . 112
nas_AddSysInfo . . . 112
nas_CallBarringSysInfo . . . 113
nas_callBarStatus . . . 114
nas_CDMAChannel . . . 115
nas_CDMAECIOThresh . . . 115
nas_CDMAInfo . . . 116
nas_CDMARSSIThresh . . . 117
nas_CDMASysInfo . . . 118
nas_CDMASysInfoExt . . . 121
nas_cellParams . . . 122
nas_CommInfo . . . 123
nas_CSGID . . . 124
nas_currentPLMN . . . 125
nas_dataSrvCapabilities . . . 126
nas_DataStatusDetail . . . 127

nas_detailSvcInfo . . . 129
nas_DeviceConfigDetail . . . 130
nas_dirNum . . . 131
nas_DRCParams . . . 132
nas_ecioListElement . . . 132
nas_ECIOThresh . . . 133
nas_errorRateListElement . . . 133
nas_GERANInfo . . . 134
nas_geranInstInfo . . . 136
nas_gsmCellInfo . . . 137
nas_GSMRSSIThresh . . . 138
nas_GSMSrvStatusInfo . . . 138
nas_GSMSysInfo . . . 139
nas_HDRECIOThresh . . . 142
nas_HDRIOThresh . . . 142
nas_HDRRSSIThresh . . . 143
nas_HDRSINRThresh . . . 144
nas_HDRSINRThreshold . . . 144
nas_HDRSysInfo . . . 145
nas_homeSIDNID . . . 147
nas_infoInterFreq . . . 148
nas_IOThresh . . . 149
nas_lteEARFCN . . . 149
nas_lteGsmCellInfo . . . 150
nas_LTEInfo . . . 151
nas_LTEInfoInterfreq . . . 154
nas_LTEInfoIntrafreq . . . 154
nas_LTEInfoNeighboringGSM . . . 156
nas_LTEInfoNeighboringWCDMA . . . 157
nas_ltePCI . . . 158
nas_lteRsrpinformation . . . 158
nas_LTERSRPThresh . . . 159
nas_LTERSRQThresh . . . 159
nas_LTERSSIThresh . . . 160
nas_LTESigRptCfg . . . 160
nas_LTESigRptConfig . . . 161
nas_lteSnrinformation . . . 162
nas_LTESNRThresh . . . 163
nas_LTESNRThreshold . . . 163
nas_LTESysInfo . . . 164
nas_lteWcdmaCellInfo . . . 166
nas_minBasedIMSI . . . 167
nas_MNRInfo . . . 168
nas_namName . . . 169
nas_netSelectionPref . . . 169
nas_NetworkStat1x . . . 170
nas_NetworkStatEVDO . . . 172
nas_nmrCellInfo . . . 174
nas_operatorNameString . . . 175
nas_OperatorPLMNData . . . 175
nas_operatorPLMNList . . . 176
nas_PhyCaAggPcellInfo . . . 177
nas_PhyCaAggScellDlBw . . . 178
nas_PhyCaAggScellIndex . . . 178
nas_PhyCaAggScellIndType . . . 179
nas_PhyCaAggScellInfo . . . 179
nas_PilotSetData . . . 182
nas_PilotSetParams . . . 182

nas_PLMNNetworkName . . . 183
nas_PLMNNetworkNameData . . . 184
nas_protocolSubtypeElement . . . 185
nas_qaQmi3Gpp2TimeZone . . . 187
nas_QmiNas3GppNetworkInfo . . . 187
nas_QmiNas3GppNetworkRAT . . . 189
nas_QmisNasPcsDigit . . . 189
nas_RejectReasonTlv . . . 190
nas_RFInfoTlv . . . 190
nas_roamIndList . . . 191
nas_RSRPThresh . . . 192
nas_rsrqInformation . . . 193
nas_RSRQThresh . . . 193
nas_RSSIThresh . . . 194
nas_rxInfo . . . 195
nas_RxSigInfo . . . 196
nas_rxSignalStrengthListElement . . . 197
nas_SccRxInfo . . . 197
nas_serviceProviderName . . . 198
nas_servSystem . . . 199
nas_sidNid . . . 201
nas_SignalStrengthTlv . . . 201
nas_SLQSSignalStrengthsIndReq . . . 201
nas_SLQSSignalStrengthsInformation . . . 203
nas_SLQSSignalStrengthsTlv . . . 204
nas_SrvStatusInfo . . . 205
nas_sysInfoCommon . . . 205
nas_TDSCDMAECIOThresh . . . 208
nas_TDSCDMARSCPThresh . . . 208
nas_TDSCDMARSSIThresh . . . 209
nas_TDSCDMASINRCONFThresh . . . 209
nas_TDSCDMASINRThresh . . . 210
nas_timeInfo . . . 210
nas_trueIMSI . . . 212
nas_txInfo . . . 213
nas_UMTSInfo . . . 213
nas_UMTSinstInfo . . . 215
nas_umtsLTENbrCell . . . 216
nas_UniversalTime . . . 217
nas_wcdmaCellInfo . . . 218
nas_WCDMAECIOThresh . . . 219
nas_WCDMAInfoLTENeighborCell . . . 219
nas_WCDMARSSIThresh . . . 220
nas_WCDMASysInfo . . . 221
nas_wcdmaUARFCN . . . 224
NASBandPreferenceTlv . . . 225
NASEmergencyModeTlv . . . 225
NasGetLTECphyCaInfo . . . 225
NASGWAcqOrderPrefTlv . . . 226
NASLTEBandPreferenceTlv . . . 226
NASLteNasReleaseInfoTlv . . . 227
NASModePreferenceTlv . . . 227
NASNetSelPreferenceTlv . . . 228
NASOTAMessageTlv . . . 228
NASPhyCaAggPcellInfo . . . 229
NASPhyCaAggScellDlBw . . . 230
NASPhyCaAggScellIndex . . . 230
NASPhyCaAggScellIndType . . . 231

NASPhyCaAggScellInfo . . . 232
NASPRLPreferenceTlv . . . 233
NASQmiCbkNasSwiOTAMessageInd . . . 233
NASQmiCbkNasSystemSelPrefInd . . . 234
NASRoamPreferenceTlv . . . 235
NASServDomainPrefTlv . . . 236
NASServingSystemInfo . . . 236
NASTimeInfoTlv . . . 237
newMTMessageTlv . . . 238
pack_cat_CATSendEnvelopeCommand_t . . . 238
pack_cat_CATSendTerminalResponse_t . . . 239
pack_cat_SetCATEventCallback_t . . . 240
pack_dms_ActivateAutomatic_t . . . 241
pack_dms_GetCustFeaturesV2_t . . . 241
pack_dms_ResetToFactoryDefaults_t . . . 242
pack_dms_SetActivationStatusCallback_t . . . 242
pack_dms_SetCrashAction_t . . . 242
pack_dms_SetCustFeature_t . . . 243
pack_dms_SetCustFeaturesV2_t . . . 245
pack_dms_SetEventReport_t . . . 246
pack_dms_SetPower_t . . . 246
pack_dms_SetUSBComp_t . . . 247
pack_dms_SLQSDmsSwiIndicationRegister_t . . . 248
pack_dms_SLQSSwiGetCrashInfo_t . . . 248
pack_dms_SLQSSwiSetDyingGaspCfg_t . . . 249
pack_dms_SLQSSwiSetHostDevInfo_t . . . 249
pack_dms_SLQSSwiSetOSInfo_t . . . 250
pack_dms_UIMChangePIN_t . . . 250
pack_dms_UIMGetControlKeyStatus_t . . . 251
pack_dms_UIMGetICCID_t . . . 252
pack_dms_UIMSetControlKeyProtection_t . . . 252
pack_dms_UIMSetPINProtection_t . . . 253
pack_dms_UIMUnblockControlKey_t . . . 253
pack_dms_UIMUnblockPIN_t . . . 254
pack_dms_UIMVerifyPIN_t . . . 255
pack_fms_GetImagesPreference_t . . . 255
pack_fms_GetStoredImages_t . . . 255
pack_fms_SetImagesPreference_t . . . 256
pack_loc_Delete_Assist_Data_t . . . 257
pack_loc_EventRegister_t . . . 258
pack_loc_SetExtPowerState_t . . . 260
pack_loc_SetOperationMode_t . . . 260
pack_loc_SLQSLOCGetBestAvailPos_t . . . 261
pack_loc_SLQSLOCGetServer_t . . . 262
pack_loc_SLQSLOCInjectPosition_t . . . 262
pack_loc_SLQSLOCInjectSensorData_t . . . 267
pack_loc_SLQSLOCInjectUTCTime_t . . . 269
pack_loc_SLQSLOCSetCradleMountConfig_t . . . 269
pack_loc_Start_t . . . 270
pack_loc_Stop_t . . . 272
pack_nas_InitiateDomainAttach_t . . . 272
pack_nas_SetACCOLC_t . . . 273
pack_nas_SetCDMANetworkParameters_t . . . 273
pack_nas_SetNetworkPreference_t . . . 275
pack_nas_SLQSConfigSigInfo_t . . . 276
pack_nas_SLQSGetPLMNName_t . . . 277
pack_nas_SLQSInitiateNetworkRegistration_t . . . 278
pack_nas_SLQSNasConfigSigInfo2_t . . . 279

pack_nas_SLQSNasGet3GPP2Subscription_t . . . 283
pack_nas_SLQSNasGetTxRxInfo_t . . . 284
pack_nas_SLQSNasIndicationRegisterExt_t . . . 284
pack_nas_SLQSNasSwiIndicationRegister_t . . . 287
pack_nas_SLQSNASSwiSetChannelLock_t . . . 288
pack_nas_SLQSSetBandPreference_t . . . 289
pack_nas_SLQSSetSignalStrengthsCallback_t . . . 291
pack_nas_SLQSSetSysSelectionPref_t . . . 291
pack_nas_SLQSSwiPSDetach_t . . . 297
pack_pds_PDSInjectTimeReference_t . . . 297
pack_pds_ResetPDSData_t . . . 298
pack_pds_SetEventReportCallback_t . . . 298
pack_pds_SetPDSDefaults_t . . . 299
pack_pds_SetPDSState_t . . . 300
pack_pds_SetPortAutomaticTracking_t . . . 300
pack_pds_SetServiceAutomaticTracking_t . . . 301
pack_pds_SetXTRAAutomaticDownload_t . . . 301
pack_pds_SetXTRANetwork_t . . . 302
pack_pds_SLQSGetAGPSConfig_t . . . 302
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t . . . 303
pack_pds_SLQSPDSInjectPositionData_t . . . 303
pack_pds_SLQSSetAGPSConfig_t . . . 305
pack_pds_SLQSSetPositionMethodState_t . . . 306
pack_pds_StartPDSTrackingSessionExt_t . . . 307
pack_qmi_t . . . 308
pack_qos_SLQSQosSwiReadApnExtraParams_t . . . 309
pack_qos_SLQSQosSwiReadDataStats_t . . . 309
pack_qos_SLQSSetQosEventCallback_t . . . 310
pack_rms_SetSMSWake_t . . . 310
pack_sar_SLQSSetRfSarState_t . . . 311
pack_sms_SaveSMS_t . . . 311
pack_sms_SendSMS_t . . . 312
pack_sms_SetNewSMSCallback_t . . . 313
pack_sms_SetSMSCAddress_t . . . 313
pack_sms_SLQSDeleteSMS_t . . . 314
pack_sms_SLQSGetSMS_t . . . 314
pack_sms_SLQSGetSmsBroadcastConfig_t . . . 315
pack_sms_SLQSGetSMSList_t . . . 315
pack_sms_SLQSModifySMSStatus_t . . . 316
pack_sms_SLQSSendAsyncSMS_t . . . 317
pack_sms_SLQSSetIndicationRegister_t . . . 317
pack_sms_SLQSSetSmsBroadcastActivation_t . . . 318
pack_sms_SLQSSetSmsBroadcastConfig_t . . . 318
pack_sms_SLQSSetSmsStorage_t . . . 319
pack_sms_SLQSSmsGetMaxStorageSize_t . . . 319
pack_sms_SLQSSmsSetRoutes_t . . . 320
pack_swiavms_SLQSAVMSSendSelection_t . . . 320
pack_swiavms_SLQSAVMSSetSettings_t . . . 321
pack_swiavms_SLQSAVMSStartSession_t . . . 323
pack_swiavms_SLQSAVMSStopSession_t . . . 323
pack_swiloc_SwiLocSetAutoStart_t . . . 324
pack_swioma_SLQSOMADMCancelSession_t . . . 325
pack_swioma_SLQSOMADMGetSessionInfo_t . . . 326
pack_swioma_SLQSOMADMSendSelection_t . . . 326
pack_swioma_SLQSOMADMSetSettings_t . . . 327
pack_swioma_SLQSOMADMStartSession_t . . . 328
pack_uim_ChangePin_t . . . 329
pack_uim_ReadTransparent_t . . . 330

pack_uim_SetPinProtection_t . . . 331
pack_uim_SLQSUIMAuthenticate_t . . . 332
pack_uim_SLQSUIMDepersonalization_t . . . 332
pack_uim_SLQSUIMEventRegister_t . . . 333
pack_uim_SLQSUIMGetConfiguration_t . . . 333
pack_uim_SLQSUIMGetFileAttributes_t . . . 334
pack_uim_SLQSUIMPowerDown_t . . . 334
pack_uim_SLQSUIMPowerUp_t . . . 335
pack_uim_SLQSUIMRefreshComplete_t . . . 335
pack_uim_SLQSUIMRefreshGetLastEvent_t . . . 336
pack_uim_SLQSUIMRefreshOK_t . . . 336
pack_uim_SLQSUIMRefreshRegister_t . . . 337
pack_uim_SLQSUIMSwitchSlot_t . . . 337
pack_uim_UnblockPin_t . . . 338
pack_uim_VerifyPin_t . . . 339
pack_voice_AnswerUSSD_t . . . 340
pack_voice_OriginateUSSD_t . . . 341
pack_voice_SLQSOriginateUSSD_t . . . 341
pack_voice_SLQSVoiceALSSelectLine_t . . . 342
pack_voice_SLQSVoiceALSSetLineSwitching_t . . . 342
pack_voice_SLQSVoiceAnswerCall_t . . . 342
pack_voice_SLQSVoiceBindSubscription_t . . . 343
pack_voice_SLQSVoiceBurstDTMF_t . . . 343
pack_voice_SLQSVoiceDialCall_t . . . 344
pack_voice_SLQSVoiceEndCall_t . . . 346
pack_voice_SLQSVoiceGetCallBarring_t . . . 346
pack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 347
pack_voice_SLQSVoiceGetCallInfo_t . . . 347
pack_voice_SLQSVoiceGetCallWaiting_t . . . 348
pack_voice_SLQSVoiceGetConfig_t . . . 348
pack_voice_SLQSVoiceIndicationRegister_t . . . 350
pack_voice_SLQSVoiceManageCalls_t . . . 351
pack_voice_SLQSVoiceOrigUSSDNoWait_t . . . 352
pack_voice_SLQSVoiceSendFlash_t . . . 352
pack_voice_SLQSVoiceSetCallBarringPassword_t . . . 353
pack_voice_SLQSVoiceSetConfig_t . . . 354
pack_voice_SLQSVoiceSetPreferredPrivacy_t . . . 355
pack_voice_SLQSVoiceSetSUPSService_t . . . 355
pack_voice_SLQSVoiceStartContDTMF_t . . . 358
pack_voice_SLQSVoiceStopContDTMF_t . . . 359
pack_wds_DHCPv4ClientLeaseChange_t . . . 359
pack_wds_GetDefaultProfile_t . . . 359
pack_wds_GetDefaultProfileNum_t . . . 360
pack_wds_GetDormancyState_t . . . 360
pack_wds_GetLastMobileIPError_t . . . 360
pack_wds_GetMobileIP_t . . . 361
pack_wds_GetMobileIPProfile_t . . . 361
pack_wds_GetPacketStatistics_t . . . 361
pack_wds_GetPacketStatus_t . . . 362
pack_wds_GetSessionDuration_t . . . 362
pack_wds_RMSetTransferStatistics_t . . . 362
pack_wds_SetAutoconnect_t . . . 363
pack_wds_SetDefaultProfile_t . . . 363
pack_wds_SetDefaultProfileNum_t . . . 365
pack_wds_SetMobileIP_t . . . 366
pack_wds_SetMobileIPParameters_t . . . 366
pack_wds_SetMobileIPProfile_t . . . 367
pack_wds_SLQSCreateProfile_t . . . 369

pack_wds_SLQSDeleteProfile_t . . . 370
pack_wds_SLQSGetCurrDataSystemStat_t . . . 370
pack_wds_SLQSGetDataBearerTechnology_t . . . 371
pack_wds_SLQSGetDUNCallInfo_t . . . 371
pack_wds_SLQSGetProfileSettings_t . . . 372
pack_wds_SLQSGetRuntimeSettings_t . . . 373
pack_wds_SLQSModifyProfile_t . . . 374
pack_wds_SLQSSet3GPPConfigItem_t . . . 375
pack_wds_SLQSSetIPFamilyPreference_t . . . 377
pack_wds_SLQSSetWdsEventCallback_t . . . 377
pack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 378
pack_wds_SLQSSSetDHCPv4ClientConfig_t . . . 378
pack_wds_SLQSSSetLoopback_t . . . 379
pack_wds_SLQSStartDataSession_t . . . 379
pack_wds_SLQSStopDataSession_t . . . 380
pack_wds_SLQSWdsSetEventReport_t . . . 381
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 383
PackCreateProfileOut . . . 383
packgetDyingGaspCfg . . . 384
packgetDyingGaspStatistics . . . 384
qmiSmsMessageList . . . 385
qmiWDSDataBearerTechnology . . . 385
RFBandInfoElements . . . 386
rmTrasnferStaticsReq . . . 387
sensorData_t . . . 388
slot_t . . . 390
slotInf . . . 391
slots_t . . . 392
sms_BroadcastConfig . . . 392
sms_CDMABroadcastConfig . . . 393
sms_getIndicationReg . . . 394
sms_getMsgWaitingInfo . . . 395
sms_getTransLayerInfo . . . 395
sms_getTransNWRegInfo . . . 396
sms_maxStorageSizeReq . . . 397
sms_maxStorageSizeResp . . . 397
sms_messageWaitingInfoContent . . . 398
sms_msgProtocolResp . . . 399
sms_qaQmi3GPP2BroadcastCfgInfo . . . 399
sms_qaQmi3GPPBroadcastCfgInfo . . . 400
sms_routeEntry . . . 400
sms_sendAsyncsmsParams . . . 401
sms_setIndicationReg . . . 404
sms_setRoutesReq . . . 404
sms_transLayerInfo . . . 405
sMSCAddress . . . 406
sMSCAddressTlv . . . 406
sMSEtwsMessage . . . 407
sMSEtwsMessageTlv . . . 407
sMSEtwsPlmn . . . 408
sMSMessageMode . . . 408
sMSMTMessage . . . 408
sMSOnIMS . . . 409
sMSOnIMSTlv . . . 409
sMSTransferRouteMTMessage . . . 410
tdscdmaSigInfoExt . . . 410
tempData_t . . . 411
transferRouteMessageTlv . . . 412

transferStatInd . . . 413
uim_appStatus . . . 413
uim_authenticateResult . . . 416
uim_authenticationData . . . 417
uim_cardResult . . . 418
uim_cardStatus . . . 418
uim_changeUIMPIN . . . 419
uim_depersonalizationInformation . . . 420
uim_encryptedPIN1 . . . 421
uim_fileAttributes . . . 422
uim_fileInfo . . . 425
uim_hotSwapStatus . . . 426
uim_personalizationStatus . . . 426
uim_readResult . . . 427
uim_readTransparentInfo . . . 428
uim_refreshevent . . . 428
uim_registerRefresh . . . 430
uim_remainingRetries . . . 431
uim_sessionInformation . . . 432
uim_setPINProtection . . . 432
uim_slotInfo . . . 433
uim_UIMSessionInformation . . . 435
uim_unblockUIMPIN . . . 436
uim_verifyUIMPIN . . . 437
unpack_cat_SetCatEventCallback_ind_t . . . 437
unpack_cat_SetCATEventCallback_t . . . 438
unpack_dms_GetActivationState_t . . . 438
unpack_dms_GetBandCapability_t . . . 439
unpack_dms_GetCrashAction_t . . . 441
unpack_dms_GetCustFeature_t . . . 441
unpack_dms_GetCustFeaturesV2_t . . . 443
unpack_dms_GetDeviceCap_t . . . 444
unpack_dms_GetDeviceCapabilities_t . . . 445
unpack_dms_GetDeviceHardwareRev_t . . . 446
unpack_dms_GetDeviceMfr_t . . . 447
unpack_dms_GetDeviceSerialNumbers_t . . . 448
unpack_dms_GetFirmwareInfo_t . . . 449
unpack_dms_GetFirmwareRevision_t . . . 451
unpack_dms_GetFirmwareRevisions_t . . . 451
unpack_dms_GetFSN_t . . . 452
unpack_dms_GetHardwareRevision_t . . . 453
unpack_dms_GetIMSI_t . . . 453
unpack_dms_GetManufacturer_t . . . 454
unpack_dms_GetModelID_t . . . 454
unpack_dms_GetNetworkTime_t . . . 455
unpack_dms_GetOfflineReason_t . . . 456
unpack_dms_GetPower_t . . . 456
unpack_dms_GetPRLVersion_t . . . 457
unpack_dms_GetSerialNumbers_t . . . 458
unpack_dms_GetUSBComp_t . . . 459
unpack_dms_GetVoiceNumber_t . . . 460
unpack_dms_ResetToFactoryDefaults_t . . . 461
unpack_dms_SetActivationStatusCallback_t . . . 462
unpack_dms_SetCrashAction_t . . . 462
unpack_dms_SetCustFeature_t . . . 462
unpack_dms_SetCustFeaturesV2_t . . . 463
unpack_dms_SetEventReport_ind_t . . . 463
unpack_dms_SetEventReport_t . . . 464

unpack_dms_SetFirmwarePreference_t . . . 464
unpack_dms_SetPower_t . . . 465
unpack_dms_SetUSBComp_t . . . 465
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t . . . 465
unpack_dms_SLQSDmsSwiGetResetInfo_t . . . 466
unpack_dms_SLQSDmsSwiIndicationRegister_t . . . 467
unpack_dms_SLQSGetBandCapability_t . . . 468
unpack_dms_SLQSGetERIFile_t . . . 471
unpack_dms_SLQSSwiClearDyingGaspStatistics_t . . . 471
unpack_dms_SLQSSwiGetCrashInfo_t . . . 472
unpack_dms_SLQSSwiGetDyingGaspCfg_t . . . 472
unpack_dms_SLQSSwiGetDyingGaspStatistics_t . . . 473
unpack_dms_SLQSSwiGetFirmwareCurr_t . . . 473
unpack_dms_SLQSSwiGetFwUpdateStatus_t . . . 474
unpack_dms_SLQSSwiGetHostDevInfo_t . . . 476
unpack_dms_SLQSSwiGetOSInfo_t . . . 477
unpack_dms_SLQSSwiGetSerialNoExt_t . . . 477
unpack_dms_SLQSSwiSetDyingGaspCfg_t . . . 478
unpack_dms_SLQSSwiSetHostDevInfo_t . . . 478
unpack_dms_SLQSSwiSetOSInfo_t . . . 478
unpack_dms_SLQSUIMGetState_t . . . 479
unpack_dms_UIMGetControlKeyStatus_t . . . 479
unpack_dms_UIMGetICCID_t . . . 480
unpack_dms_UIMGetPINStatus_t . . . 481
unpack_dms_UIMSetControlKeyProtection_t . . . 482
unpack_dms_UIMSetPINProtection_t . . . 483
unpack_dms_UIMUnblockControlKey_t . . . 484
unpack_fms_GetImagesPreference_t . . . 484
unpack_fms_GetStoredImages_t . . . 485
unpack_fms_SetImagesPreference_t . . . 485
unpack_loc_BestAvailPos_Ind_t . . . 486
unpack_loc_Delete_Assist_Data_t . . . 491
unpack_loc_DeleteAssistData_Ind_t . . . 492
unpack_loc_EngineState_Ind_t . . . 493
unpack_loc_EventNMEA_Ind_t . . . 493
unpack_loc_EventRegister_t . . . 494
unpack_loc_GetServer_Ind_t . . . 494
unpack_loc_GnssSvInfo_Ind_t . . . 496
unpack_loc_PositionRpt_Ind_t . . . 496
unpack_loc_SetExtPowerConfig_Ind_t . . . 503
unpack_loc_SetExtPowerState_t . . . 503
unpack_loc_SetOperationMode_Ind_t . . . 504
unpack_loc_SetOperationMode_t . . . 504
unpack_loc_SLQSLOCGetBestAvailPos_t . . . 505
unpack_loc_Start_t . . . 505
unpack_loc_Stop_t . . . 505
unpack_nas_GetCDMANetworkParameters_t . . . 506
unpack_nas_GetHomeNetwork3GPP2_t . . . 508
unpack_nas_GetHomeNetwork_t . . . 510
unpack_nas_GetNetworkPreference_t . . . 511
unpack_nas_GetRFInfo_t . . . 512
unpack_nas_GetServingNetwork_t . . . 513
unpack_nas_GetServingNetworkCapabilities_t . . . 515
unpack_nas_GetSignalStrengths_t . . . 516
unpack_nas_PerformNetworkScan_t . . . 517
unpack_nas_SetDataCapabilitiesCallback_ind_t . . . 518
unpack_nas_SetEventReportInd_t . . . 518
unpack_nas_SetNasLTECphyCaIndCallback_ind_t . . . 519

unpack_nas_SetNetworkPreference_t . . . 520
unpack_nas_SetRoamingIndicatorCallback_ind_t . . . 520
unpack_nas_SetServingSystemCallback_ind_t . . . 521
unpack_nas_SLQSGetErrorRate_t . . . 521
unpack_nas_SlqsGetLTECphyCAInfo_t . . . 522
unpack_nas_SLQSGetNetworkTime_t . . . 523
unpack_nas_SLQSGetOperatorNameData_t . . . 523
unpack_nas_SLQSGetPLMNName_t . . . 524
unpack_nas_SLQSGetServingSystem_t . . . 525
unpack_nas_SLQSGetSignalStrength_t . . . 529
unpack_nas_SLQSGetSysInfo_t . . . 531
unpack_nas_SLQSGetSysSelectionPref_t . . . 534
unpack_nas_SLQSNasGet3GPP2Subscription_t . . . 538
unpack_nas_SLQSNasGetCellLocationInfo_t . . . 539
unpack_nas_SLQSNasGetHDRColorCode_t . . . 541
unpack_nas_SLQSNasGetSigInfo_t . . . 541
unpack_nas_SLQSNasGetTxRxInfo_t . . . 542
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t . . . 542
unpack_nas_SLQSNasSigInfoCallback_ind_t . . . 543
unpack_nas_SLQSNASSwiGetChannelLock_t . . . 544
unpack_nas_SLQSNasSwiModemStatus_t . . . 545
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t . . . 546
unpack_nas_SLQSNasTimerCallback_ind_t . . . 546
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t . . . 547
unpack_nas_SLQSSwiGetHDRPersonality_t . . . 547
unpack_nas_SLQSSwiGetHDRProtSubtype_t . . . 548
unpack_nas_SLQSSwiGetHRPDStats_t . . . 549
unpack_nas_SLQSSwiGetLteCQI_t . . . 549
unpack_nas_SLQSSwiGetLteSccRxInfo_t . . . 550
unpack_nas_SLQSSwiNetworkDebug_t . . . 550
unpack_nas_SLQSSysInfoCallback_ind_t . . . 551
unpack_omaDmConfigTlv_t . . . 554
unpack_omaDmFotaTlv_t . . . 555
unpack_omaDmNotificationsTlv_t . . . 557
unpack_pds_GetPDSDefaults_t . . . 558
unpack_pds_GetPDSState_t . . . 559
unpack_pds_GetPortAutomaticTracking_t . . . 560
unpack_pds_GetServiceAutomaticTracking_t . . . 560
unpack_pds_GetXTRAAutomaticDownload_t . . . 560
unpack_pds_GetXTRANetwork_t . . . 561
unpack_pds_GetXTRAValidity_t . . . 561
unpack_pds_SetEventReport_Ind_t . . . 562
unpack_pds_SetPdsState_Ind_t . . . 564
unpack_pds_SLQSGetAGPSConfig_t . . . 564
unpack_pds_SLQSGetGPSStateInfo_t . . . 565
unpack_qmi_t . . . 569
unpack_qos_dataRate_t . . . 570
unpack_qos_IPv4Addr_t . . . 570
unpack_qos_IPv6Addr_t . . . 571
unpack_qos_IPv6TrafCls_t . . . 571
unpack_qos_pktErrRate_t . . . 572
unpack_qos_Port_t . . . 572
unpack_qos_QosFlowInfo_t . . . 572
unpack_qos_QosFlowInfoState_t . . . 574
unpack_qos_SLQSQosGetNetworkStatus_t . . . 575
unpack_qos_SLQSQosSwiReadApnExtraParams_t . . . 575
unpack_qos_SLQSQosSwiReadDataStats_t . . . 576
unpack_qos_SLQSSetQosEventCallback_ind_t . . . 578

unpack_qos_SLQSSetQosNWStatusCallback_ind_t . . . 578
unpack_qos_SLQSSetQosPriEventCallback_ind_t . . . 579
unpack_qos_SLQSSetQosStatusCallback_ind_t . . . 579
unpack_qos_swiQosFilter_t . . . 580
unpack_qos_swiQosFlow_t . . . 584
unpack_qos_tokenBucket_t . . . 588
unpack_qos_Tos_t . . . 588
unpack_QosFlowStat_t . . . 589
unpack_rms_GetSMSWake_t . . . 590
unpack_RMTransferStatistics_ind_t . . . 590
unpack_sar_SLQSGetRfSarState_t . . . 591
unpack_sms_GetSMSCAddress_t . . . 592
unpack_sms_SaveSMS_t . . . 593
unpack_sms_SendSMS_t . . . 593
unpack_sms_SetNewSMSCallback_ind_t . . . 593
unpack_sms_SetNewSMSCallback_t . . . 595
unpack_sms_SLQSDeleteSMS_t . . . 595
unpack_sms_SLQSGetIndicationRegister_t . . . 595
unpack_sms_SLQSGetMessageWaiting_t . . . 595
unpack_sms_SLQSGetSMS_t . . . 596
unpack_sms_SLQSGetSmsBroadcastConfig_t . . . 596
unpack_sms_SLQSGetSMSList_t . . . 597
unpack_sms_SLQSGetTransLayerInfo_t . . . 597
unpack_sms_SLQSGetTransNWRegInfo_t . . . 598
unpack_sms_SLQSModifySMSStatus_t . . . 598
unpack_sms_SLQSSmsGetMaxStorageSize_t . . . 598
unpack_sms_SLQSSmsGetMessageProtocol_t . . . 599
unpack_sms_SLQSSwiGetSMSStorage_t . . . 599
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t . . . 600
unpack_swiavms_SLQSAVMSGetSettings_t . . . 600
unpack_swiavms_SLQSAVMSSendSelection_t . . . 601
unpack_swiavms_SLQSAvmsSetEventReport_t . . . 601
unpack_swiavms_SLQSAVMSSetSettings_t . . . 602
unpack_swiavms_SLQSAVMSStartSession_t . . . 602
unpack_swiavms_SLQSAVMSStopSession_t . . . 603
unpack_swiloc_SwiLocGetAutoStart_t . . . 603
unpack_swioma_SLQSOMADMAlertCallback_ind_t . . . 605
unpack_swioma_SLQSOMADMGetSessionInfo_t . . . 605
unpack_swioma_SLQSOMADMGetSettings_t . . . 608
unpack_swioma_SLQSOMADMStartSession_t . . . 610
unpack_uim_ChangePin_t . . . 610
unpack_uim_GetCardStatus_t . . . 611
unpack_uim_ReadTransparent_t . . . 611
unpack_uim_SetPinProtection_t . . . 612
unpack_uim_SetUimSlotStatusChangeCallback_ind_t . . . 613
unpack_uim_SLQSUIMAuthenticate_t . . . 614
unpack_uim_SLQSUIMDepersonalization_t . . . 614
unpack_uim_SLQSUIMEventRegister_t . . . 615
unpack_uim_SLQSUIMGetConfiguration_t . . . 615
unpack_uim_SLQSUIMGetFileAttributes_t . . . 616
unpack_uim_SLQSUIMGetSlotsStatus_t . . . 617
unpack_uim_SLQSUIMRefreshGetLastEvent_t . . . 617
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t . . . 617
unpack_uim_UnblockPin_t . . . 618
unpack_uim_VerifyPin_t . . . 618
unpack_voice_SLQSOriginateUSSD_t . . . 619
unpack_voice_SLQSVoiceAnswerCall_t . . . 620
unpack_voice_SLQSVoiceBurstDTMF_t . . . 620

unpack_voice_SLQSVoiceDialCall_t . . . 621
unpack_voice_SLQSVoiceEndCall_t . . . 621
unpack_voice_SLQSVoiceGetAllCallInfo_t . . . 622
unpack_voice_SLQSVoiceGetCallBarring_t . . . 624
unpack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 626
unpack_voice_SLQSVoiceGetCallInfo_t . . . 627
unpack_voice_SLQSVoiceGetCallWaiting_t . . . 629
unpack_voice_SLQSVoiceGetCLIP_t . . . 631
unpack_voice_SLQSVoiceGetCLIR_t . . . 632
unpack_voice_SLQSVoiceGetCNAP_t . . . 633
unpack_voice_SLQSVoiceGetCOLP_t . . . 634
unpack_voice_SLQSVoiceGetCOLR_t . . . 636
unpack_voice_SLQSVoiceGetConfig_t . . . 637
unpack_voice_SLQSVoiceManageCalls_t . . . 639
unpack_voice_SLQSVoiceSendFlash_t . . . 639
unpack_voice_SLQSVoiceSetCallBarringPassword_t . . . 640
unpack_voice_SLQSVoiceSetConfig_t . . . 641
unpack_voice_SLQSVoiceSetSUPSService_t . . . 642
unpack_voice_SLQSVoiceStartContDTMF_t . . . 643
unpack_voice_SLQSVoiceStopContDTMF_t . . . 644
unpack_wds_DHCPv4ClientLease_ind_t . . . 644
unpack_wds_GetAutoconnect_t . . . 645
unpack_wds_GetByteTotals_t . . . 645
unpack_wds_GetConnectionRate_t . . . 646
unpack_wds_GetDataBearerTechnology_t . . . 647
unpack_wds_GetDefaultProfile_t . . . 647
unpack_wds_GetDefaultProfileNum_t . . . 649
unpack_wds_GetDormancyState_t . . . 650
unpack_wds_GetLastMobileIPError_t . . . 650
unpack_wds_GetMobileIP_t . . . 651
unpack_wds_GetMobileIPProfile_t . . . 651
unpack_wds_GetPacketStatistics_t . . . 653
unpack_wds_GetPacketStatus_t . . . 654
unpack_wds_GetSessionDuration_t . . . 656
unpack_wds_GetSessionState_t . . . 657
unpack_wds_RMSetTransferStatistics_t . . . 657
unpack_wds_SetMobileIPProfile_t . . . 657
unpack_wds_SLQSCreateProfile_t . . . 657
unpack_wds_SLQSDeleteProfile_t . . . 658
unpack_wds_SLQSGet3GPPConfigItem_t . . . 658
unpack_wds_SLQSGetCurrDataSystemStat_t . . . 660
unpack_wds_SLQSGetCurrentChannelRate_t . . . 660
unpack_wds_SLQSGetDataBearerTechnology_t . . . 661
unpack_wds_SLQSGetDUNCallInfo_t . . . 662
unpack_wds_SLQSGetProfileSettings_t . . . 665
unpack_wds_SLQSGetRuntimeSettings_t . . . 665
unpack_wds_SLQSModifyProfile_t . . . 667
unpack_wds_SLQSSetIPFamilyPreference_t . . . 668
unpack_wds_SLQSSetPacketSrvStatusCallback_t . . . 668
unpack_wds_SLQSSetWdsEventCallback_ind_t . . . 669
unpack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 671
unpack_wds_SLQSSGetLoopback_t . . . 672
unpack_wds_SLQSStartDataSession_t . . . 672
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 673
UnPackGetProfileSettingOut . . . 675
unpackWdsProfileParam . . . 675
voice_airTimer . . . 676
voice_allCallsAlphaIDInfo . . . 677

voice_allCallsDiagInfo . . . 677
voice_allCallsUUSInfo . . . 678
voice_alphaIDInfo . . . 678
voice_arrAlertingPattern . . . 679
voice_arrAlertingType . . . 680
voice_arrAlphaID . . . 680
voice_arrCalledPartyNum . . . 681
voice_arrCallEndReason . . . 681
voice_arrCallInfo . . . 682
voice_arrConnectPartyNum . . . 683
voice_arrDiagInfo . . . 683
voice_arrRedirPartyNum . . . 684
voice_arrRemotePartyName . . . 684
voice_arrRemotePartyNum . . . 685
voice_arrSvcOption . . . 685
voice_arrUUSInfo . . . 686
voice_burstDTMFInfo . . . 686
voice_calledPartySubAdd . . . 687
voice_callFwdTypeAndPlan . . . 688
voice_callFWExtInfo . . . 689
voice_callFWInfo . . . 691
voice_callInfo . . . 692
voice_ccSUPSType . . . 694
voice_CLIPResp . . . 694
voice_CLIRResp . . . 695
voice_CNAPResp . . . 696
voice_COLPResp . . . 697
voice_COLRResp . . . 697
voice_connectNumInfo . . . 698
voice_CUGInfo . . . 700
voice_curAMRConfig . . . 700
voice_diagInfo . . . 701
voice_DTMFLengths . . . 702
voice_getAllCallInformation . . . 702
voice_getAllCallRmtPtyName . . . 703
voice_getAllCallRmtPtyNum . . . 703
voice_getCallFWExtInfo . . . 704
voice_getCallFWInfo . . . 705
voice_peerNumberInfo . . . 705
voice_prefVoiceSO . . . 707
voice_remotePartyName . . . 709
voice_remotePartyNum . . . 710
voice_roamTimer . . . 710
voice_USSInfo . . . 711
voice_UUSInfo . . . 712
wds_currNetworkInfo . . . 713
wds_DataULongLongTlv . . . 714
wds_DataULongTlv . . . 715
wds_DHCPLeaseOptTlv . . . 715
wds_DHCPLeaseStateTlv . . . 716
wds_DHCPOpt . . . 716
wds_DHCPProfileIdTlv . . . 717
wds_DHCPv4HWConfig . . . 717
wds_DHCPv4Option . . . 718
wds_DHCPv4OptionList . . . 719
wds_DHCPv4ProfileId . . . 719
wds_Domain . . . 720
wds_DomainNameList . . . 720

wds_GPRSQoS 721
wds_IPv4AdTlv 721
wds_IPV6AddressInfo 722
wds_IPV6GWAddressInfo 722
wds_PCSCFFQDNAddress 723
wds_PCSCFFQDNAddressList 723
wds_PCSCFIPv4ServerAddressList 724
wds_ProfileIdentifier 724
wds_profileInfo 725
wds_TrStatInd 725
wds_UMTSMinQoS 726
wdsDhcpv4HwConfig 728
wdsDhcpv4Option 729
wdsDhcpv4OptionList 730
wdsDhcpv4ProfileId 730

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

apdoxypages.c
Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages . . . 731
cat.h . . . 731
common.h . . . 734
dms.h . . . 739
fms.h . . . 792
lite-fw.h . . . 795
loc.h . . . 803
nas.h . . . 819
pds.h . . . 864
qaGobiApiTableBandClasses.h
Network Access Service API Band Classes table . . . 884
qaGobiApiTableCallControlReturnReasons.h
Call Control Return Reasons table . . . 887
qaGobiApiTableCallEndReasons.h
Wireless Data Service Call End Reasons . . . 888
qaGobiApiTableCarrierCodes.h
Carrier Codes table . . . 904
qaGobiApiTableCodingScheme.h
Data Coding Scheme . . . 906
qaGobiApiTableGpsCapabilityCodes.h
Position Determination Service API GPS Capability Codes . . . 909
qaGobiApiTablePowerModes.h
Device Management Service API Power Modes table . . . 910
qaGobiApiTableRadioInterfaces.h
Network Access Service API Radio Interfaces table . . . 910
qaGobiApiTableRegionCodes.h
Region Codes table . . . 911
qaGobiApiTableServiceOptions.h
Voice Service Options . . . 911
qaGobiApiTableSupServiceInfoClasses.h
Voice Supplementary Service Information Classes . . . 914
qaGobiApiTableSwiAudio.h
Swi Audio related tables . . . 914
qaGobiApiTableSwiOMADMSessionStatus.h
Session Status table . . . 915

qaGobiApiTableSwiOMADMUpdateCompleteStatus.h
Update Complete Status table . . . 916
qaGobiApiTableVoiceCallEndReasons.h
Voice Service Call and supplementary services end reasons . . . 917
qmerrno.h . . . 924
qos.h . . . 931
rms.h . . . 939
sar.h . . . 941
sms.h . . . 943
swiavms.h . . . 965
SwiDataTypes.h
SWI data types . . . 983
swiloc.h . . . 984
swioma.h . . . 986
SWIWWANCMAPI.h . . . 995
uim.h . . . 995
voice.h . . . 1013
wds.h . . . 1040

Chapter 6

Module Documentation

6.1 QMI pack/unpack (lite-qmi)

Files

- file common.h
- file dms.h
- file loc.h
- file nas.h
- file qos.h
- file sms.h
- file swioma.h
- file uim.h
- file wds.h
- file fms.h
- file swiloc.h
- file pds.h
- file rms.h
- file sar.h
- file swiavms.h
- file voice.h
- file cat.h

6.1.1 Detailed Description

6.2 Streaming Download Protocol (lite-fw)

Files

- file lite-fw.h

6.2.1 Detailed Description

Chapter 7

Namespace Documentation

7.1 Tables Namespace Reference

7.1.1 Detailed Description

Tables referenced in the API function headers:

- Table 1 - Call End Reason Codes The reason a call (either in process or connected) was ended. qaGobiApiTableCallEndReasons.h
- Table 2 - Carrier codes List of carrier identification codes. qaGobiApiTableCarrierCodes.h
- Table 3 - Region codes List of region identification codes. qaGobiApiTableRegionCodes.h
- Table 4 - GPS capability codes List of GPS capability codes. qaGobiApiTableGpsCapabilityCodes.h
- Table 5 - Radio Interfaces List of radio interface technologies. qaGobiApiTableRadioInterfaces.h
- Table 6 - Band classes List of band classes. qaGobiApiTableBandClasses.h
- Table 7 - Power modes List of operating modes. qaGobiApiTablePowerModes.h
- Table 8 - Service Options List of Service Options. qaGobiApiTableServiceOptions.h
- Table 9 - Voice Call End Reason List of Voice Call End Reason. qaGobiApiTableVoiceCallEndReasons.h
- Table 10 - Data Coding Scheme List of Data Coding Scheme. qaGobiApiTableCodingScheme.h
- Table 11 - Call Control Return Reasons List of Voice Call Control Return Reasons. qaGobiApiTableCallControlReturnReasons.h
- Table 12 - Supplementary Service Information Classes List of Voice Supplementary Service Information Classes. qaGobiApiTableSupServiceInfoClasses.h
- Table 13 - Audio Calibration Data Base(ACDB) Device List of ACDB devices. qaGobiApiTableSwiAudio.h
- Table 14 - PIFACE List of physical interfaces. qaGobiApiTableSwiAudio.h

Chapter 8

Data Structure Documentation

8.1 _litefw_FirmwareInfo_ Struct Reference

Data Fields

- char szModelid_str [20]
- char szFwversion_str [20]
- char szSku_str [20]
- char szPackageid_str [20]
- char szCarrier_str [20]
- char szCarrierPriversion_str [20]

8.1.1 Detailed Description

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

szModelid_str	• Model Name String
szFwversion_str	• Firmware Version String.
szSku_str	• SKU String.
szPackageid_str	• Package ID String.
szCarrier_str	• Carrier String.
szCarrier-Priversion_str	• Carrier PRI Version String.

8.1.2 Field Documentation

8.1.2.1 char _litefw_FirmwareInfo_::szCarrier_str[20]

8.1.2.2 char _litefw_FirmwareInfo_::szCarrierPriversion_str[20]

8.1.2.3 char _litefw_FirmwareInfo_::szFwversion_str[20]

8.1.2.4 char _litefw_FirmwareInfo_::szModelid_str[20]

8.1.2.5 char _litefw_FirmwareInfo_::szPackageid_str[20]

8.1.2.6 char _litefw_FirmwareInfo_::szSku_str[20]

8.2 _PackSwiAvmsSetSettingsAPNInfo Struct Reference

Data Fields

- uint8_t bAPNLength
- uint8_t szAPN [49]
- uint8_t bUnameLength
- uint8_t szUname [29]
- uint8_t bPWDLength
- uint8_t szPWD [29]

8.2.1 Detailed Description

Parameters

bAPNLength	APN Length.
szAPN	APN String.
bUnameLength	Username Length.
szUname	User Name String.
bPWDLength	Password Length.
szPWD	Password String.

8.2.2 Field Documentation

8.2.2.1 uint8_t _PackSwiAvmsSetSettingsAPNInfo::bAPNLength

8.2.2.2 uint8_t _PackSwiAvmsSetSettingsAPNInfo::bPWDLength

8.2.2.3 uint8_t _PackSwiAvmsSetSettingsAPNInfo::bUnameLength

8.2.2.4 uint8_t _PackSwiAvmsSetSettingsAPNInfo::szAPN[49]

8.2.2.5 uint8_t _PackSwiAvmsSetSettingsAPNInfo::szPWD[29]

8.2.2.6 uint8_t _PackSwiAvmsSetSettingsAPNInfo::szUname[29]

8.3 _PackSwiAvmsSetSettingsConnectionRetryTimers Struct Reference

Data Fields

- uint16_t Timers [8]

8.3.1 Field Documentation

8.3.1.1 uint16_t _PackSwiAvmsSetSettingsConnectionRetryTimers::Timers[8]

8.4 _PackSwiAvmsSetSettingsPeriodInfo Struct Reference

Data Fields

- uint32_t ulMin
- uint32_t ulMax

8.4.1 Detailed Description

Parameters

ulMin	Min period in seconds.
ulMax	Max period in seconds.

8.4.2 Field Documentation

8.4.2.1 uint32_t _PackSwiAvmsSetSettingsPeriodInfo::ulMax

8.4.2.2 uint32_t _PackSwiAvmsSetSettingsPeriodInfo::ulMin

8.5 _PackSwiAVMSSettingsAPNInfo Struct Reference

Data Fields

- uint8_t bAPNLength
- uint8_t * pAPN
- uint8_t bUnameLength
- uint8_t * pUname
- uint8_t bPWDLength
- uint8_t * pPWD

8.5.1 Field Documentation

8.5.1.1 uint8_t _PackSwiAVMSSettingsAPNInfo::bAPNLength

8.5.1.2 uint8_t _PackSwiAVMSSettingsAPNInfo::bPWDLength

8.5.1.3 uint8_t _PackSwiAVMSSettingsAPNInfo::bUnameLength

8.5.1.4 uint8_t* _PackSwiAVMSSettingsAPNInfo::pAPN

8.5.1.5 uint8_t* _PackSwiAVMSSettingsAPNInfo::pPWD

8.5.1.6 uint8_t* _PackSwiAVMSSettingsAPNInfo::pUname

8.6 _PackSwiAVMSSettingsConnectionRetryTimers Struct Reference

Data Fields

- uint16_t Timers [8]

8.6.1 Field Documentation

8.6.1.1 uint16_t _PackSwiAVMSSettingsConnectionRetryTimers::Timers[8]

8.7 _PackSwiAVMSSettingsPeriodsInfo Struct Reference

Data Fields

- uint32_t min
- uint32_t max

8.7.1 Field Documentation

8.7.1.1 uint32_t _PackSwiAVMSSettingsPeriodsInfo::max

8.7.1.2 uint32_t _PackSwiAVMSSettingsPeriodsInfo::min

8.8 _unpack_swiavms_SLQSAVMSEventReportInd_t Struct Reference

Data Fields

- UnpackSwiAvmsEventReportBinaryUpdateSessionInfo * pBinaryUpdateSessionInfo
- UnpackSwiAvmsEventReportConfig * pConfig
- UnpackSwiAvmsEventReportNotification * pNotification
- UnpackSwiAvmsEventReportPackageID * pPackageID
- UnpackSwiAvmsEventReportConnectionRequest * pConnectionRequest
- UnpackSwiAvmsEventReportWAMSParamChange * pWAMSParaChanged
- UnpackSwiAvmsEventReportRegStatus * pRegStatus
- UnpackSwiAvmsEventReportDataSessionStatus * pDataSessionStatus
- UnpackSwiAvmsEventReportSessionType * pSessionType
- UnpackSwiAvmsEventReportHTTPStatus * pHTTPStatus
- uint32_t resultcode

8.8.1 Detailed Description

Structure that contains the session type response for AVMS event report indication command

Parameters

pBinaryUpdateSessionInfo[OUT]	• See UnpackSwiAvmsEventReportBinaryUpdateSessionInfo for more information
pNotification[OUT]	• See UnpackSwiAvmsEventReportConfig for more information
pPackageID[OUT]	• See UnpackSwiAvmsEventReportPackageID for more information

pConnection-Request[OUT]	• See UnpackSwiAvmsEventReportConnectionRequest for more information
pWAMSPara-Changed[OUT]	• See UnpackSwiAvmsEventReportWAMSParamChange for more information
pRegStatus[OU-T]	• See UnpackSwiAvmsEventReportRegStatus for more information
pDataSession-Status[OUT]	• See UnpackSwiAvmsEventReportDataSessionStatus for more information
pSessionType[-OUT]	• See UnpackSwiAvmsEventReportSessionType for more information
pWAMSPara-Changed[OUT]	• See UnpackSwiAvmsEventReportWAMSParamChange for more information
pHTTPStatus[O-UT]	• See UnpackSwiAvmsEventReportHTTPStatus for more information
resultcode	- Tlv Result Code.

8.8.2 Field Documentation

8.8.2.1 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pBinaryUpdateSessionInfo

8.8.2.2 UnpackSwiAvmsEventReportConfig∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pConfig

8.8.2.3 UnpackSwiAvmsEventReportConnectionRequest∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pConnectionRequest

8.8.2.4 UnpackSwiAvmsEventReportDataSessionStatus∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pDataSessionStatus

8.8.2.5 UnpackSwiAvmsEventReportHTTPStatus∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pHTTPStatus

8.8.2.6 UnpackSwiAvmsEventReportNotification∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pNotification

8.8.2.7 UnpackSwiAvmsEventReportPackageID∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pPackageID

8.8.2.8 UnpackSwiAvmsEventReportRegStatus∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pRegStatus

8.8.2.9 UnpackSwiAvmsEventReportSessionType∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pSessionType

8.8.2.10 UnpackSwiAvmsEventReportWAMSParamChange∗ _unpack_swiavms_SLQSAVMSEventReportInd_t::pWAMSParaChanged

8.8.2.11 uint32_t _unpack_swiavms_SLQSAVMSEventReportInd_t::resultcode

8.9 _unpack_swiavms_SLQSAVMSSessionGetInfo_t Struct Reference

Data Fields

- UnpackSwiAvmsEventReportBinaryUpdateSessionInfo * pBinaryUpdateSessionInfo
- UnpackSwiAvmsEventReportConfig * pConfig
- UnpackSwiAvmsEventReportNotification * pNotification
- UnpackSwiAvmsEventReportPackageID * pPackageID
- uint32_t resultcode

8.9.1 Detailed Description

Structure that contains the session type response for AVMS get session info command

Parameters

resultcode	- Tlv Result Code.

8.9.2 Field Documentation

8.9.2.1 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo* _unpack_swiavms_SLQSAVMSSessionGetInfo_t::pBinaryUpdateSessionInfo

8.9.2.2 UnpackSwiAvmsEventReportConfig* _unpack_swiavms_SLQSAVMSSessionGetInfo_t::pConfig

8.9.2.3 UnpackSwiAvmsEventReportNotification* _unpack_swiavms_SLQSAVMSSessionGetInfo_t::pNotification

8.9.2.4 UnpackSwiAvmsEventReportPackageID* _unpack_swiavms_SLQSAVMSSessionGetInfo_t::pPackageID

8.9.2.5 uint32_t _unpack_swiavms_SLQSAVMSSessionGetInfo_t::resultcode

8.10 _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo Struct Reference

Data Fields

- uint8_t bBinaryType
- uint8_t bState
- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint32_t ulPkgDownloadSize
- uint32_t ulPkgDownloadComplete
- uint16_t wUpdateCompeteStatus
- uint8_t bSerity
- uint16_t wVersionLength
- uint8_t szVersion [128]
- uint16_t wNameLength
- uint8_t szName [128]
- uint16_t wDescriptionLength
- uint8_t szDescription [1024]
- uint8_t TlvPresent

8.10.1 Detailed Description

Parameters

bBinaryType	: Type • 1 - Firmware. • 2 - User App. • 3 - Legato Framework.
bStat	: State • 0x01 - No binary update available. • 0x02 - Query binary Download. • 0x03 - Binary Downloading. • 0x04 - Binary downloaded. • 0x05 -Query Binary Update. • 0x06 - Binary updating. • 0x07 - Binary updated.
bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
wUserInput-Timeout	Timeout for user input in minutes. A value of 0 means no time-out.
ulPkgDownload-Size	The size (in bytes) of the update package
ulPkgDownload-Complete	The number of bytes being downloaded. For downloading state, this value shall be $>$ 0 and incremented toward the pkg_dload_size. For other states, the value shall be 0 as it is meaningless.
wUpdate-CompeteStatus	Result code.This field should be looked at only when the AVMS session is complete.
bSerity	Serity. • 0x01 - Mandatory. • 0x02 - Optional.
wVersionLength	Length of FW Version string in bytes.
szVersion	FW Version string in ASCII.
wNameLength	Length Package Name string in bytes.
szName	Package Name in UCS2.
wDescription-Length	Length of description in bytes.
szDescription	Description of Update Package in USC2.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.10.2 Field Documentation

8.10.2.1 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bBinaryType

8.10.2.2 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bSerity

8.10.2.3 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bState

8.10.2.4 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::bUserInputRequest

8.10.2.5 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szDescription[1024]

8.10.2.6 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szName[128]

8.10.2.7 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::szVersion[128]

8.10.2.8 uint8_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::TlvPresent

8.10.2.9 uint32_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::ulPkgDownloadComplete

8.10.2.10 uint32_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::ulPkgDownloadSize

8.10.2.11 uint16_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wDescriptionLength

8.10.2.12 uint16_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wNameLength

8.10.2.13 uint16_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wUpdateCompeteStatus

8.10.2.14 uint16_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wUserInputTimeout

8.10.2.15 uint16_t _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo::wVersionLength

8.11 _UnpackSwiAvmsEventReportConfig Struct Reference

Data Fields

- uint8_t bState
- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint16_t wAlertMsgLength
- uint8_t szAlertMsg [200]
- uint8_t TlvPresent

8.11.1 Detailed Description

Parameters

bState	State. • 0x01 - AVMS Read Request. • 0x02 - AVMS Change Request. • 0x03 - AVMS Config Complete.
bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
wUserInput-Timeout	Timeout for user input in seconds. A value of 0 means no time-out
wAlertMsg-Length	Length of Alert message string in bytes.
szAlertMsg	Alert message in UCS2.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.11.2 Field Documentation

8.11.2.1 uint8_t _UnpackSwiAvmsEventReportConfig::bState

8.11.2.2 uint8_t _UnpackSwiAvmsEventReportConfig::bUserInputRequest

8.11.2.3 uint8_t _UnpackSwiAvmsEventReportConfig::szAlertMsg[200]

8.11.2.4 uint8_t _UnpackSwiAvmsEventReportConfig::TlvPresent

8.11.2.5 uint16_t _UnpackSwiAvmsEventReportConfig::wAlertMsgLength

8.11.2.6 uint16_t _UnpackSwiAvmsEventReportConfig::wUserInputTimeout

8.12 _UnpackSwiAvmsEventReportConnectionRequest Struct Reference

Data Fields

- uint8_t bUserInputRequest
- uint16_t wUserInputTimeout
- uint8_t TlvPresent

8.12.1 Detailed Description

Parameters

bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
bUserInput-Request	Timeout for user input in minutes. A value of 0 means no time-out.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.12.2 Field Documentation

8.12.2.1 uint8_t _UnpackSwiAvmsEventReportConnectionRequest::bUserInputRequest

8.12.2.2 uint8_t _UnpackSwiAvmsEventReportConnectionRequest::TlvPresent

8.12.2.3 uint16_t _UnpackSwiAvmsEventReportConnectionRequest::wUserInputTimeout

8.13 _UnpackSwiAvmsEventReportDataSessionStatus Struct Reference

Data Fields

- uint8_t bType
- uint16_t wErrorCode
- uint8_t TlvPresent

8.13.1 Detailed Description

Parameters

bType	Notification type. • 0: Data session closed. • 1: Data session activated. • 2: Register made. -3: Data session error.
wErrorCode	LWM2M Session error code. • 0x0000: none
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.13.2 Field Documentation

8.13.2.1 uint8_t _UnpackSwiAvmsEventReportDataSessionStatus::bType

8.13.2.2 uint8_t _UnpackSwiAvmsEventReportDataSessionStatus::TlvPresent

8.13.2.3 uint16_t _UnpackSwiAvmsEventReportDataSessionStatus::wErrorCode

8.14 _UnpackSwiAvmsEventReportHTTPStatus Struct Reference

Data Fields

- uint16_t wHTTPStatus
- uint8_t TlvPresent

8.14.1 Detailed Description

Parameters

wHTTPStatus	See RFC 7231. • 100 - Continue • 101 - Switching Protocols • 200 - OK • 201 - Created • 202 - Accepted • 203 - Non-Authoritative Information • 204 - No Content • 205 - Reset Content • 206 - Partial Content • 300 - Multiple Choices • 301 - Moved Permanently • 302 - Found • 303 - See Other • 304 - Not Modified • 305 - Use Proxy • 307 - Temporary Redirect • 400 - Bad Request • 401 - Unauthorized • 402 - Payment Required • 403 - Forbidden • 404 - Not Found • 405 - Method Not Allowed • 406 - Not Acceptable • 407 - Proxy Authentication Required • 408 - Request Timeout • 409 - Conflict • 410 - Gone • 411 - Length Required • 412 - Precondition Failed • 413 - Payload Too Large • 414 - URI Too Long • 415 - Unsupported Media Type • 416 - Range Not Satisfiable • 417 - Expectation Failed • 426 - Upgrade Required • 500 - Internal Server Error • 501 - Not Implemented • 502 - Bad Gateway • 503 - Service Unavailable • 504 - Gateway Timeout • 505 - HTTP Version Not Supported
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.14.2 Field Documentation

8.14.2.1 uint8_t _UnpackSwiAvmsEventReportHTTPStatus::TlvPresent

8.14.2.2 uint16_t _UnpackSwiAvmsEventReportHTTPStatus::wHTTPStatus

8.15 _UnpackSwiAvmsEventReportNotification Struct Reference

Data Fields

- uint8_t bNotification
- uint16_t wSessionStatus
- uint8_t TlvPresent

8.15.1 Detailed Description

Parameters

bNotification	Notification. • 0x14 - Module starts sending data to server. • 0x15 - Authentication with the server. • 0x16 - session with the server is ended.
wSessionStatus	This field will set to the session status for notifications that occur at the end of a session, zero for all other notifications. • 0x0000: Successful: Session succeeded • 0x0001: Break: Session succeeded • 0x0002: Large Object Handled: Session succeeded • 0x0003: No status: Session succeeded • 0x0004: No more commands: Session succeeded • 0x0005: User cancel: Session cancelled • 0x0100-0x01FF: General errors • 0x0200-0x02FF: Syncml errors • 0x0300-0x03FF: Authentication errors • 0x0400-0x04FF: Protocol errors • 0x0500-0x05FF: Tree errors (DM Only) • 0x0600-0x06FF: Not applicable • 0x0700-0x07FF: Trigger errors • 0x0800-0x08FF: FUMO errors • 0x0900-0x09FF: Communication errors • 0x0A00-0x0AFF: Parsing errors • 0x0B00-0x0CFF: Not applicable • 0x7F00-0x7F12:Insignia errors • 0x7F13: Illegal text: Text received contains illegal characters • 0x7F14: Download failure: Failed to download FOTA image • 0x7F15: Empty session: Session ran successfully, but no information wasa updated • 0x7F16: Factory reset successful: Factory reset succeeded • 0x7F17: Factory reset fail: Factory reset failed
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.15.2 Field Documentation

8.15.2.1 uint8_t _UnpackSwiAvmsEventReportNotification::bNotification

8.15.2.2 uint8_t _UnpackSwiAvmsEventReportNotification::TlvPresent

8.15.2.3 uint16_t _UnpackSwiAvmsEventReportNotification::wSessionStatus

8.16 _UnpackSwiAvmsEventReportPackageID Struct Reference

Data Fields

- uint8_t bPackageID
- uint8_t TlvPresent

8.16.1 Detailed Description

Parameters

bPackageID	Package ID of the application binary that this AVMS_EVENT_ID notification is for.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.16.2 Field Documentation

8.16.2.1 uint8_t _UnpackSwiAvmsEventReportPackageID::bPackageID

8.16.2.2 uint8_t _UnpackSwiAvmsEventReportPackageID::TlvPresent

8.17 _UnpackSwiAvmsEventReportRegStatus Struct Reference

Data Fields

- uint8_t bRegStatus
- uint8_t TlvPresent

8.17.1 Detailed Description

Parameters

bRegStatus	LWM2M Registration status. • 0: Need Bootstrap. • 1: Bootstrap made. • 2: Register made. -3: Update made.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.17.2 Field Documentation

8.17.2.1 uint8_t _UnpackSwiAvmsEventReportRegStatus::bRegStatus

8.17.2.2 uint8_t _UnpackSwiAvmsEventReportRegStatus::TlvPresent

8.18 _UnpackSwiAvmsEventReportSessionType Struct Reference

Data Fields

- uint8_t bType
- uint8_t TlvPresent

8.18.1 Detailed Description

Parameters

bType	Session Type. • 0: Bootstrap session. • 1. DM session
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.18.2 Field Documentation

8.18.2.1 uint8_t _UnpackSwiAvmsEventReportSessionType::bType

8.18.2.2 uint8_t _UnpackSwiAvmsEventReportSessionType::TlvPresent

8.19 _UnpackSwiAvmsEventReportWAMSParamChange Struct Reference

Data Fields

- uint16_t wWamsChangeMask
- uint8_t TlvPresent

8.19.1 Detailed Description

Parameters

wWamsChange-Mask	Mask of WAMS parameters changed. • By default set to 0xFF for all changes. • 0x01 - device_login • 0x02 - device_MD5_key • 0x04 - server_login • 0x08 - server_MD5_key • 0x10 - server_URL • 0x20 - Nonce • 0x40 - Application key
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

8.19.2 Field Documentation

8.19.2.1 uint8_t _UnpackSwiAvmsEventReportWAMSParamChange::TlvPresent

8.19.2.2 uint16_t _UnpackSwiAvmsEventReportWAMSParamChange::wWamsChangeMask

8.20 altSrcInfo_t Struct Reference

Data Fields

- uint32_t source
- uint32_t linkage
- uint32_t coverage

8.20.1 Detailed Description

This structure specifies information regarding the altitude source

Parameters

source	• Specifies the source of the altitude • Valid values – 0 - Source is unknown – 1 - GPS is the source – 2 - Cell ID provided the source – 3 - Source is enhanced cell ID – 4 - Wi-Fi is the source – 5 - Terrestrial source – 6 - Hybrid terrestrial source – 7 - Altitude database is the source – 8 - Barometric altimeter is the source – 9 - Other sources
linkage	• Specifies the dependency between the horizontal and altitude position components • Valid values – 0 - Not specified – 1 - Fully interdependent – 2 - Depends on latitude and longitude – 3 - Fully independent
coverage	• Specifies the region of uncertainty. • Valid values – 0 - Not specified – 1 - Altitude uncertainty is valid at the injected horizontal position coordinates only – 2 - Altitude uncertainty applies to the position of the device regardless of horizontal position

8.20.2 Field Documentation

8.20.2.1 uint32_t altSrcInfo_t::coverage

8.20.2.2 uint32_t altSrcInfo_t::linkage

8.20.2.3 uint32_t altSrcInfo_t::source

8.21 appStats Struct Reference

Data Fields

- uint8_t appType
- uint8_t appState
- uint8_t persoState
- uint8_t persoFeature
- uint8_t persoRetries
- uint8_t persoUnblockRetries
- uint8_t aidLength
- uint8_t aidVal [255]
- uint8_t univPin
- uint8_t pin1State
- uint8_t pin1Retries
- uint8_t puk1Retries
- uint8_t pin2State
- uint8_t pin2Retries
- uint8_t puk2Retries

8.21.1 Detailed Description

This structure contains Application Status Information loaded on the card.

Parameters

appType	• Indicates the type of the application. – 0 - Unknown – 1 - SIM card – 2 - USIM application – 3 - RUIM card – 4 - CSIM application – 5 - ISIM application • Other values are reserved for the future and are to be handled as "Unknown".
appState	• Indicates the state of the application. – 0 - Unknown – 1 - Detected – 2 - PIN1 or UPIN is required – 3 - PUK1 or PUK for UPIN is required – 4 - Personalization state must be checked – 5 - PIN1 is blocked – 6 - Illegal – 7 - Ready
persoState	• Indicates the state of the personalization for the application. – 0 - Unknown – 1 - Personalization operation is in progress – 2 - Ready – 3 - Personalization code is required – 4 - PUK for personalization code is required – 5 - Permanently blocked

persoFeature	• Indicates the personalization feature. • This applies only when a personalization code is required to deactivate or unblock personalization. – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization – 11 - Unknown
persoRetries	• Indicates the number of retries remaining to disable the personalization.
persoUnblock-Retries	• Indicates the number of retries remaining to unblock the personalization.
aidLength	• Number of sets of the following elements. i.e. aidVal • If zero(0) then no aidVal information exists.
aidVal[MAX_DESCRIPTION_LENGTH]	• Application identifier value.
univPin	• Indicates whether UPIN replaces PIN1. – 0 - PIN1 is used – 1 - UPIN replaces PIN1
pin1State	• Indicates the state of PIN1. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin1Retries	• Indicates the number of retries remaining to verify PIN1.
puk1Retries	• Indicates the number of retries remaining to unblock PIN1.

pin2State	• Indicates the state of PIN2. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin2Retries	• Indicates the number of retries remaining to verify PIN2.
puk2Retries	• Indicates the number of retries remaining to unblock PIN2.

8.21.2 Field Documentation

8.21.2.1 uint8_t appStats::aidLength

8.21.2.2 uint8_t appStats::aidVal[255]

8.21.2.3 uint8_t appStats::appState

8.21.2.4 uint8_t appStats::appType

8.21.2.5 uint8_t appStats::persoFeature

8.21.2.6 uint8_t appStats::persoRetries

8.21.2.7 uint8_t appStats::persoState

8.21.2.8 uint8_t appStats::persoUnblockRetries

8.21.2.9 uint8_t appStats::pin1Retries

8.21.2.10 uint8_t appStats::pin1State

8.21.2.11 uint8_t appStats::pin2Retries

8.21.2.12 uint8_t appStats::pin2State

8.21.2.13 uint8_t appStats::puk1Retries

8.21.2.14 uint8_t appStats::puk2Retries

8.21.2.15 uint8_t appStats::univPin

8.22 CarrierImage_t Struct Reference

Data Fields

- uint32_t m_nCarrierId

- uint32_t m_nFolderId
- uint32_t m_nStorage
- uint8_t m_FwImageId [16]
- uint8_t m_FwBuildId [32]
- uint8_t m_PriImageId [16]
- uint8_t m_PriBuildId [32]

8.22.1 Detailed Description

This structure contains the Carrier Image parameters.

Parameters

m_nCarrierId	• Unique numeric carrier ID indicating the carrier that the following images belong to
m_nFolderId	• Unique numeric folder ID indicating the folder where the images should reside on the host storage.
m_nStorage	• Information of storage type • Values – 0 - Device – 1 - Host
m_FwImageId	• Firmware image ID
m_FwBuildId	• Firmware build ID
m_PriImageId	• PRI image ID
m_PriBuildId	• PRI build ID

8.22.2 Field Documentation

8.22.2.1 uint8_t CarrierImage_t::m_FwBuildId[32]

8.22.2.2 uint8_t CarrierImage_t::m_FwImageId[16]

8.22.2.3 uint32_t CarrierImage_t::m_nCarrierId

8.22.2.4 uint32_t CarrierImage_t::m_nFolderId

8.22.2.5 uint32_t CarrierImage_t::m_nStorage

8.22.2.6 uint8_t CarrierImage_t::m_PriBuildId[32]

8.22.2.7 uint8_t CarrierImage_t::m_PriImageId[16]

8.23 cat_AlPhaIdentifierTlv Struct Reference

Data Fields

- uint8_t ReferenceID
- uint16_t AlphaIDLength
- uint8_t AlphaID [255]

8.23.1 Detailed Description

structure used to store all Alpha Identifier parameters.

Parameters

ReferenceID	- proactive command type that included the alpha identifier - 0x01; sends SMS proactive command
AlphaIDLength	- length of AlphaID (in bytes)
AlphaID	- alpha identifier, encoded as in ETSI TS 102 223 [Section 8.2]

8.23.2 Field Documentation

8.23.2.1 uint8_t cat_AlPhaIdentifierTlv::AlphaID[255]

8.23.2.2 uint16_t cat_AlPhaIdentifierTlv::AlphaIDLength

8.23.2.3 uint8_t cat_AlPhaIdentifierTlv::ReferenceID

8.24 cat_commonEventTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EventID
- uint16_t EventLength
- union cat_currentCatEvent CatEvent

8.24.1 Detailed Description

Structure used to store all Common CAT Event TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
EventID	- Event ID. Can be any of the following 16 - Display Text 17 - Get In-Key 18 - Get Input 19 - Setup Menu 20 - Select Item 21 - Send SMS - Alpha Identifier 22 - Setup Event List 23 - Setup Idle Mode Text 24 - Language Notification 25 - Refresh 26 - End Proactive Session
EventLength	- Length of pData (in Bytes)
CatEvent	- Structure to the Data specific to the CAT event ID

8.24.2 Field Documentation

8.24.2.1 union cat_currentCatEvent cat_commonEventTlv::CatEvent

8.24.2.2 uint8_t cat_commonEventTlv::EventID

8.24.2.3 uint16_t cat_commonEventTlv::EventLength

8.24.2.4 uint8_t cat_commonEventTlv::TlvPresent

8.25 cat_currentCatEvent Union Reference

Data Fields

- struct cat_EventIDDataTlv CatEvIDData
- struct cat_AlPhaIdentifierTlv CatAlphaIdtfr
- struct cat_EventListTlv CatEventLst
- struct cat_RefreshTlv CatRefresh
- struct cat_EndProactiveSessionTlv CatEndPS

8.25.1 Detailed Description

Union used to represent the current CAT Event Data. Choose the structure based on the EventID received.

- Use cat_EventIDDataTlv if the Event ID is any of the below.
  - 16
  - 17
  - 18
  - 19
  - 20
  - 23
  - 24
- Use cat_AlPhaIdentifierTlv if the Event ID is 21
- Use cat_EventListTlv if the Event ID is 22
- Use cat_RefreshTlv if the Event ID is 25
- Use cat_EndProactiveSessionTlv if the Event ID is 26

8.25.2 Field Documentation

8.25.2.1 struct cat_AlPhaIdentifierTlv cat_currentCatEvent::CatAlphaIdtfr

8.25.2.2 struct cat_EndProactiveSessionTlv cat_currentCatEvent::CatEndPS

8.25.2.3 struct cat_EventListTlv cat_currentCatEvent::CatEventLst

8.25.2.4 struct cat_EventIDDataTlv cat_currentCatEvent::CatEvIDData

8.25.2.5 struct cat_RefreshTlv cat_currentCatEvent::CatRefresh

8.26 cat_EndProactiveSessionTlv Struct Reference

Data Fields

- uint8_t EndProactiveSession

8.26.1 Detailed Description

structure used to store End Proactive Session event parameters.

Parameters

EndProactive-Session	- The proactive session end type values are: • 0x01 - End proactive session command type received from the card • 0x02 - End proactive session internal to ME

8.26.2 Field Documentation

8.26.2.1 uint8_t cat_EndProactiveSessionTlv::EndProactiveSession

8.27 cat_EventIDDataTlv Struct Reference

Data Fields

- uint32_t ReferenceID
- uint16_t DataLength
- uint8_t Data [255]

8.27.1 Detailed Description

structure used to store all Common CAT Event parameters.

Parameters

ReferenceID	- proactive command reference ID.
DataLength	- length of pData (in Bytes)
Data	- command specific to the CAT event ID, encoded as in ETSI TS 102 223 [Section 6.6.X]

8.27.2 Field Documentation

8.27.2.1 uint8_t cat_EventIDDataTlv::Data[255]

8.27.2.2 uint16_t cat_EventIDDataTlv::DataLength

8.27.2.3 uint32_t cat_EventIDDataTlv::ReferenceID

8.28 cat_EventListTlv Struct Reference

Data Fields

- uint32_t SetupEventList

8.28.1 Detailed Description

structure used to store all Event List parameters.

Parameters

SetupEventList	- Setup event list bit mask • 0x00000001 - User Activity Notify • 0x00000002 - Idle Screen Available • 0x00000004 - Lang Selection Notify Each set bit indicates the availability of the corresponding event in Setup Event list proactive command; all unlisted bits are reserved for future use and will be ignored

8.28.2 Field Documentation

8.28.2.1 uint32_t cat_EventListTlv::SetupEventList

8.29 cat_RefreshTlv Struct Reference

Data Fields

- uint16_t RefreshMode
- uint8_t RefreshStage

8.29.1 Detailed Description

structure used to store all Refresh Event parameters.

Parameters

RefreshMode	- The Refresh Event as in ETSI TS 102 223 [Section 8.6]
RefreshStage	- Stage of a refresh procedure • 0x01 - Refresh start • 0x02 - Refresh success • 0x03 - Refresh failed

8.29.2 Field Documentation

8.29.2.1 uint16_t cat_RefreshTlv::RefreshMode

8.29.2.2 uint8_t cat_RefreshTlv::RefreshStage

8.30 cdmaSSInfo Struct Reference

Data Fields

- int8_t rssi
- int16_t ecio

8.30.1 Detailed Description

This structure contains the parameters for CDMA/WCDMA Signal Strength Information

Parameters

rssi	• RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal.
ecio	• ECIO value representing negative 0.5 dBm increments, i.e., 2 means -1 dBm (14 means -7 dBm, 63 means -31.5 dBm).

8.30.2 Field Documentation

8.30.2.1 int16_t cdmaSSInfo::ecio

8.30.2.2 int8_t cdmaSSInfo::rssi

8.31 connectionStatus Struct Reference

Data Fields

- uint8_t MDMConnStatus
- uint64_t MDMCallDuration

8.31.1 Detailed Description

This structure contains modem connection status

Parameters

MDMConn-Status	• Current link status – 0x01 - DISCONNECTED – 0x02 - CONNECTED
MDMCall-Duration	• Call duration in milliseconds. • If the modem connection status is connected, this represent the duration of the current DUN call. • If the modem connection status is disconnected, this represents the duration of the last DUN call since the device was powered up (zero, if no call has been made or if the last call was not DUN).

8.31.2 Field Documentation

8.31.2.1 uint64_t connectionStatus::MDMCallDuration

8.31.2.2 uint8_t connectionStatus::MDMConnStatus

8.32 crashInfoParams Struct Reference

Data Fields

- uint8_t crashStatus
- crashInformation crashInfo

8.32.1 Detailed Description

This structure contains crash information parameters

Parameters

crashStatus[OUT]	• Device Crash Status • 0 - no crash • 1 - crash has occurred
crashInfo[OUT]	• See crashInformation

8.32.2 Field Documentation

8.32.2.1 crashInformation crashInfoParams::crashInfo

8.32.2.2 uint8_t crashInfoParams::crashStatus

8.33 crashInformation Struct Reference

Data Fields

- uint16_t numCrashes
- uint32_t crashId
- uint32_t crashData
- uint16_t crashStrlen
- char crashString [255]
- uint16_t gcdumpStrlen
- char gcdumpString [1024]

8.33.1 Detailed Description

This structure contains crash information

Parameters

numCrashes[OUT]	• number of instances of the remaining fields
crashId[OUT]	• ramdom crash id assigned at crash

crashData	• crash data[OUT]
crashStrlen[IN/-OUT]	• length of the crashString field as an input, length of the crashString field returned by API as an OUTPUT
crashString	• crash string
gcdumpStrlen[O-UT]	• length of the gcdumpString field as an input, length of the gcdumpString field returned by API as an OUTPUT
gcdumpString[O-UT]	• gcdump string for the crash

8.33.2 Field Documentation

8.33.2.1 uint32_t crashInformation::crashData

8.33.2.2 uint32_t crashInformation::crashId

8.33.2.3 char crashInformation::crashString[255]

8.33.2.4 uint16_t crashInformation::crashStrlen

8.33.2.5 char crashInformation::gcdumpString[1024]

8.33.2.6 uint16_t crashInformation::gcdumpStrlen

8.33.2.7 uint16_t crashInformation::numCrashes

8.34 currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.34.1 Detailed Description

Network information structure

Parameters

NetworkType	• Values: – 0 - 3GPP – 1 - 3GPP2

RATMask	• Radio Access Technology (RAT) mask to indicate the type of technology. • Values: – 0 - Don't Care – 0x8000 - NULL Bearer • CDMA RAT mask values: – 0x01 - CDMA_1x – 0x02 - EVDO_REV0 – 0x04 - EVDO_REVA – 0x08 - EVDO_REVB – 0x10 - EHRPD – 0x20 - FMC • UMTS RAT mask values: – 0x01 - WCDMA – 0x02 - GPRS – 0x04 - HSDPA – 0x08 - HSUPA – 0x10 - EDGE – 0x20 - LTE – 0x40 - HSDPA+ – 0x80 - DC_HSDPA+ – 0x100 - 64_QAM – 0x200 - TDSCDMA
SOMask	• Service Option (SO) mask to indicate the service option or type of application. • Values: – 0 - Don't Care • CDMA 1x SO mask values: – 0x01 - CDMA_1X_IS95 – 0x02 - CDMA_1X_IS2000 – 0x04 - CDMA_1X_IS2000_REL_A • CDMA EV-DO Rev 0 SO mask values: – 0x01 - DPA • CDMA EV-DO Rev A SO mask values: – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD • CDMA EV-DO Rev B SO mask values: – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD – 0x10 - MMPA – 0x20 - MMPA_EHRPD

8.34.2 Field Documentation

8.34.2.1 uint8_t currNetworkInfo::NetworkType

8.34.2.2 uint32_t currNetworkInfo::RATMask

8.34.2.3 uint32_t currNetworkInfo::SOMask

8.35 dms_ActivationStatusTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t activationStatus

8.35.1 Detailed Description

Activation Status Tlv

Parameters

TlvPresent	• Tlv Present
activationStatus	• Service Activation Code – 0 - Service not activated – 1 - Service activated – 2 - Activation connecting – 3 - Activation connected – 4 - OTASP security authenticated – 5 - OTASP NAM downloaded – 6 - OTASP MDN downloaded – 7 - OTASP IMSI downloaded – 8 - OTASP PRL downloaded – 9 - OTASP SPC downloaded – 10 - OTASP settings committed

8.35.2 Field Documentation

8.35.2.1 uint32_t dms_ActivationStatusTlv::activationStatus

8.35.2.2 uint16_t dms_ActivationStatusTlv::TlvPresent

8.36 dms_OperatingModeTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t operatingMode

8.36.1 Detailed Description

Operating Mode Tlv

Parameters

TlvPresent	• Tlv Present
operatingMode	• 0 - Online • 1 - Low power • 2 - Factory test mode • 3 - Offline • 4 - Resetting • 5 - Shutting down • 6 - Persistent low power • 7 - Mode-only low power

8.36.2 Field Documentation

8.36.2.1 uint32_t dms_OperatingModeTlv::operatingMode

8.36.2.2 uint16_t dms_OperatingModeTlv::TlvPresent

8.37 DMScustSettingInfo Struct Reference

Data Fields

- uint16_t id_length
- uint8_t cust_id [64+1]
- uint16_t value_length
- uint8_t cust_value [8+1]
- uint16_t cust_attr

8.37.1 Detailed Description

This structure contains information about Customization Setting. This TLV is only applicable for 9x30 modules so far

Parameters

id_length	• length of cust_id field
cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.
value_length	• length of cust_value field

cust_value	• Customization Setting Value (Maximum 8 bytes)
cust_attr	• Customization Setting attribute through QMI – bit 0: Values: * 0 - read only * 1 - read/write

8.37.2 Field Documentation

8.37.2.1 uint16_t DMScustSettingInfo::cust_attr

8.37.2.2 uint8_t DMScustSettingInfo::cust_id[64+1]

8.37.2.3 uint8_t DMScustSettingInfo::cust_value[8+1]

8.37.2.4 uint16_t DMScustSettingInfo::id_length

8.37.2.5 uint16_t DMScustSettingInfo::value_length

8.38 DMScustSettingList Struct Reference

Data Fields

- uint8_t list_type
- uint16_t num_instances
- DMScustSettingInfo custSetting [255+1]

8.38.1 Detailed Description

This structure contains the fields of TLV Customization Setting List. This TLV is only applicable for 9x30 modules so far

Parameters

list_type	• list type requested
num_instances	• number of instances of customization setting
custSetting	• See DMScustSettingInfo for more information

8.38.2 Field Documentation

8.38.2.1 DMScustSettingInfo DMScustSettingList::custSetting[255+1]

8.38.2.2 uint8_t DMScustSettingList::list_type

8.38.2.3 uint16_t DMScustSettingList::num_instances

8.39 DMSgetCustomFeatureV2 Struct Reference

Data Fields

- DMSgetCustomInput * pGetCustomInput
- DMScustSettingInfo * pCustSettingInfo
- DMScustSettingList * pCustSettingList

8.39.1 Detailed Description

This struture contains the TLV required to get the Customization Info and customization list.

Parameters

pGetCustomInput[IN]	• See DMSgetCustomInput for more information
pCustSettingInfo[OUT]	• See DMScustSettingInfo for more information
pCustSettingList[OUT]	• See DMScustSettingList for more information

8.39.2 Field Documentation

8.39.2.1 DMScustSettingInfo* DMSgetCustomFeatureV2::pCustSettingInfo

8.39.2.2 DMScustSettingList* DMSgetCustomFeatureV2::pCustSettingList

8.39.2.3 DMSgetCustomInput* DMSgetCustomFeatureV2::pGetCustomInput

8.40 DMSgetCustomInput Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint8_t list_type

8.40.1 Detailed Description

This structure contains which customization id or the list type want to retrieve from modem. This TLV is only applicable for 9x30 modules so far

Parameters

cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.

list_type	• list type requested

8.40.2 Field Documentation

8.40.2.1 uint8_t DMSgetCustomInput::cust_id[64+1]

8.40.2.2 uint8_t DMSgetCustomInput::list_type

8.41 dunchannelRate Struct Reference

Data Fields

- uint32_t CurrChanTxRate
- uint32_t CurrChanRxRate
- uint32_t MaxChanTxRate
- uint32_t MaxChanRxRate

8.41.1 Detailed Description

This structure contains Channel Rate

Parameters

CurrChanTxRate	• Instantaneous channel Tx rate in bits per second
CurrChanRx-Rate	• Instantaneous channel Rx rate in bits per second
MaxChanTxRate	• maximum Tx rate that can be assigned to the device by the serving system in bits per second
MaxChanRx-Rate	• maximum Rx rate that can be assigned to the device by the serving system in bits per second

8.41.2 Field Documentation

8.41.2.1 uint32_t dunchannelRate::CurrChanRxRate

8.41.2.2 uint32_t dunchannelRate::CurrChanTxRate

8.41.2.3 uint32_t dunchannelRate::MaxChanRxRate

8.41.2.4 uint32_t dunchannelRate::MaxChanTxRate

8.42 eriDataparams Struct Reference

Data Fields

- uint16_t eriDataLen
- uint8_t eriData [1024]

8.42.1 Detailed Description

This structure contains Extended Roaming Indicator(ERI) file parameters

Parameters

eriDataLen	• Upon input, the maximum number of bytes that file contents array can contain. • Upon successful output, actual number of bytes written to file contents array
eriData	• ERI data read from persistent storage(Max size is 1024)

8.42.2 Field Documentation

8.42.2.1 uint8_t eriDataparams::eriData[1024]

8.42.2.2 uint16_t eriDataparams::eriDataLen

8.43 eTWSPLMNInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsPlmnInfo ETWSPLMNInfo

8.43.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
ETWSPLMNInfo	• ETWS PLMN Information • See sMSEtwsPlmnInfo for more information

8.43.2 Field Documentation

8.43.2.1 sMSEtwsPlmnInfo eTWSPLMNInfoTlv::ETWSPLMNInfo

8.43.2.2 uint8_t eTWSPLMNInfoTlv::TlvPresent

8.44 FMSImageElement Struct Reference

Data Fields

- uint8_t imageType
- uint8_t imageId [16]
- uint8_t buildIdLength
- uint8_t buildId [100]

8.44.1 Detailed Description

This structure contains the Image Element information

Parameters

imageType	• Type of image 0 - Modem 1 - PRI
imageId	• Unique image identifier
buildIdLength	• Length of the build ID string (may be zero)
pBuildId	• Build ID ANSI string with length provided by the previous field

8.44.2 Field Documentation

8.44.2.1 uint8_t FMSImageElement::buildId[100]

8.44.2.2 uint8_t FMSImageElement::buildIdLength

8.44.2.3 uint8_t FMSImageElement::imageId[16]

8.44.2.4 uint8_t FMSImageElement::imageType

8.45 FMSImageIdElement Struct Reference

Data Fields

- uint8_t storageIndex
- uint8_t failureCount
- uint8_t imageID [16]
- uint8_t buildIDLength
- uint8_t buildID [100]

8.45.1 Detailed Description

This structure contains the Image ID list element Information

Parameters

storageIndex	• Index in storage where the image is located(a value of 0xFF indicates that the storage for this type of image is not relevant)
failureCount	• Number of consecutive write attempts to this storage index that have failed(a value of 0xFF indicates unspecified)
imageID	• Image unique identifier(max 16 chars.)
buildIDLength	• Length of the build ID string. If there is no build ID, this field will be 0 and no data will follow.
buildID	• String containing image build information(Max 100 characters)

8.45.2 Field Documentation

8.45.2.1 uint8_t FMSImageIdElement::buildID[100]

8.45.2.2 uint8_t FMSImageIdElement::buildIDLength

8.45.2.3 uint8_t FMSImageIdElement::failureCount

8.45.2.4 uint8_t FMSImageIdElement::imageID[16]

8.45.2.5 uint8_t FMSImageIdElement::storageIndex

8.46 FMSImageIDEntries Struct Reference

Data Fields

- uint8_t imageType
- uint8_t maxImages
- uint8_t executingImage
- uint8_t imageIDSize
- FMSImageIdElement imageIDElement [50]

8.46.1 Detailed Description

This structure contains the list entry Information

Parameters

imageType	• Type of image – 0 - Modem – 1 - PRI

maxImages	• Maximum number of images of this type that may be stored concurrently on the device
executingImage	• Index (into the next array) of image that is currently executing
imageIDSize	• The number of elements in the image ID list
imageIDElement	• Array of ImageIDElement Structure (Max 50 elements) • See FMSImageIdElement

8.46.2 Field Documentation

8.46.2.1 uint8_t FMSImageIDEntries::executingImage

8.46.2.2 FMSImageIdElement FMSImageIDEntries::imageIDElement[50]

8.46.2.3 uint8_t FMSImageIDEntries::imageIDSize

8.46.2.4 uint8_t FMSImageIDEntries::imageType

8.46.2.5 uint8_t FMSImageIDEntries::maxImages

8.47 FMSImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageIDEntries imageIDEntries [2]

8.47.1 Detailed Description

This structure contains the Get Stored Images List

Parameters

listSize	• The number of elements in the image list
imageIDEntries	• Array of ImageIDEntries Structure (Max 2 entries)

8.47.2 Field Documentation

8.47.2.1 FMSImageIDEntries FMSImageList::imageIDEntries[2]

8.47.2.2 uint8_t FMSImageList::listSize

8.48 FMSPrefImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageElement listEntries [2]

8.48.1 Detailed Description

This structure contains the Preference Image List information

Parameters

listSize	• The number of elements in the image list
pListEntries	• Array of Image entries with size provided by previous field • See FMSImageElement

8.48.2 Field Documentation

8.48.2.1 FMSImageElement FMSPrefImageList::listEntries[2]

8.48.2.2 uint8_t FMSPrefImageList::listSize

8.49 hdrSSInfo Struct Reference

Data Fields

- int8_t rssi
- int16_t ecio
- uint8_t sinr
- int32_t io

8.49.1 Detailed Description

This structure contains the parameters for HDR Signal Strength Information

Parameters

rssi	• RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal.
ecio	• ECIO value representing negative 0.5 dBm increments, i.e., 2 means -1 dBm (14 means -7 dBm, 63 means -31.5 dBm).

sinr	• SINR level. • SINR is only applicable for 1xEV-DO. • Valid levels are 0 to 8, where the maximum value for: – 0 - SINR_LEVEL_0 is -9 dB – 1 - SINR_LEVEL_1 is -6 dB – 2 - SINR_LEVEL_2 is -4.5 dB – 3 - SINR_LEVEL_3 is -3 dB – 4 - SINR_LEVEL_4 is -2 dB – 5 - SINR_LEVEL_5 is +1 dB – 6 - SINR_LEVEL_6 is +3 dB – 7 - SINR_LEVEL_7 is +6 dB – 8 - SINR_LEVEL_8 is +9 dB – 0xFF - Not Available
io	• Received IO in dBm. • IO is only applicable for 1xEV-DO.

8.49.2 Field Documentation

8.49.2.1 int16_t hdrSSInfo::ecio

8.49.2.2 int32_t hdrSSInfo::io

8.49.2.3 int8_t hdrSSInfo::rssi

8.49.2.4 uint8_t hdrSSInfo::sinr

8.50 image_info_t Struct Reference

Data Fields

- uint8_t imageType
- uint8_t uniqueID [16]
- uint8_t buildIDLen
- uint8_t buildID [255]

8.50.1 Detailed Description

This structure is used to store image information

Parameters

imageType	• Image Type • Values: – 0 - FW – 1 - configuration

uniqueID	• Image Unique Identifier (ASCII characters)
buildIDLen	• Length of build ID string to follow • If set to zero, build ID string will be blank
buildID	• String containing image information(ASCII characters) • Maximum length of this string is 255 chars

8.50.2 Field Documentation

8.50.2.1 uint8_t image_info_t::buildID[255]

8.50.2.2 uint8_t image_info_t::buildIDLen

8.50.2.3 uint8_t image_info_t::imageType

8.50.2.4 uint8_t image_info_t::uniqueID[16]

8.51 ipv6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.51.1 Detailed Description

This structure contains the IPv6 address information

Parameters

IPV6PrefixLen	Length of the received IPv6 address
IPAddressV6	IPv6 address(in network byte order)

8.51.2 Field Documentation

8.51.2.1 uint16_t ipv6AddressInfo::IPAddressV6[8]

8.51.2.2 uint8_t ipv6AddressInfo::IPV6PrefixLen

8.52 LibPackGPRSRequestedQoS Struct Reference

Data Fields

- uint32_t precedenceClass
- uint32_t delayClass
- uint32_t reliabilityClass
- uint32_t peakThroughputClass

- uint32_t meanThroughputClass

8.52.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

Parameters

precedence-Class	• Precedence class
delayClass	• Delay class
reliabilityClass	• Reliability class
peak-Throughput-Class	• Peak throughput class
mean-Throughput-Class	• Mean throughput class

8.52.2 Field Documentation

8.52.2.1 uint32_t LibPackGPRSRequestedQoS::delayClass

8.52.2.2 uint32_t LibPackGPRSRequestedQoS::meanThroughputClass

8.52.2.3 uint32_t LibPackGPRSRequestedQoS::peakThroughputClass

8.52.2.4 uint32_t LibPackGPRSRequestedQoS::precedenceClass

8.52.2.5 uint32_t LibPackGPRSRequestedQoS::reliabilityClass

8.53 LibpackProfile3GPP Struct Reference

Data Fields

- uint8_t * pProfilename
- uint16_t * pProfilenameSize
- uint8_t * pPDPtype
- uint8_t * pPdpHdrCompType
- uint8_t * pPdpDataCompType
- uint8_t * pAPNName
- uint16_t * pAPNnameSize
- uint32_t * pPriDNSIPv4AddPref
- uint32_t * pSecDNSIPv4AddPref
- LibPackUMTSQoS * pUMTSReqQoS
- LibPackUMTSQoS * pUMTSMinQoS
- LibPackGPRSRequestedQoS * pGPRSRequestedQos
- LibPackGPRSRequestedQoS * pGPRSMinimumQoS
- uint8_t * pUsername

- uint16_t * pUsernameSize
- uint8_t * pPassword
- uint16_t * pPasswordSize
- uint8_t * pAuthenticationPref
- uint32_t * pIPv4AddrPref
- uint8_t * pPcscfAddrUsingPCO
- uint8_t * pPdpAccessConFlag
- uint8_t * pPcscfAddrUsingDhcp
- uint8_t * pImCnFlag
- LibPackTFTIDParams * pTFTID1Params
- LibPackTFTIDParams * pTFTID2Params
- uint8_t * pPdpContext
- uint8_t * pSecondaryFlag
- uint8_t * pPrimaryID
- uint16_t * pIPv6AddPref
- LibPackUMTSReqQoSSigInd * pUMTSReqQoSSigInd
- LibPackUMTSReqQoSSigInd * pUMTSMinQosSigInd
- uint16_t * pPriDNSIPv6addpref
- uint16_t * pSecDNSIPv6addpref
- uint8_t * pAddrAllocPref
- LibPackQosClassID * pQosClassID
- uint8_t * pAPNDisabledFlag
- uint32_t * pPDNInactivTimeout
- uint8_t * pAPNClass

8.53.1 Detailed Description

This structure contains Input parameters of unpack_wds_SLQSGetProfileSettings_t Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pProfileName	• One or more uint8_ts describing the profile
pProfilename-Size;	• This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 uint8_ts.
pPDPType	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0x00 - PDP-IP (IPv4) – 0x01 - PDP-PPP – 0x02 - PDP-IPV6 – 0x03 - PDP-IPV4V6
pPdpHdrComp-Type	• PDP header compression type – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095

pPdpDataCompType	• PDP data compression type – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression
pAPNName	• Access point name
pAPNnameSize;	• This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 uint8_ts.
pPriDNSIPv4AddPref	• Primary DNS IPv4 Address Preference
pSecDNSIPv4AddPref	• Secondary DNS IPv4 Address Preference
pUMTSReqQoS	• UMTS Requested QoS
pUMTSMinQoS	• UMTS Minimum QoS
pGPRSRequestedQoS	• GPRS Requested QoS
pGPRSMinimumQoS	• GPRS Minimum QoS
pUsername	• User name
pUsernameSize;	• This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 uint8_ts.
pPassword	• Password
pPasswordSize;	• This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 uint8_ts.

pAuthentication-Pref	• Authentication Preference – Bit map that indicates the authentication algorithm preference * Bit 0 - PAP preference · 0 - PAP is never performed · 1 - PAP may be performed * Bit 1 - CHAP preference · 0 - CHAP is never performed · 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism.
pIPv4AddrPref	• IPv4 Address Preference
pPcscfAddr-UsingPCO	• P-CSCF Address using PCO Flag – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0
pPdpAccess-ConFlag	• PDP access control flag – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission
pPcscfAddr-UsingDhcp	• P-CSCF address using DHCP – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0
pImCnFlag	• IM CN flag – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile
pTFTID1Params	• Traffic Flow Template
pTFTID2Params	• Traffic Flow Template
pPdpContext	• PDP context number
pSecondaryFlag	• PDP context secondary flag – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile

pPrimaryID	• PDP context primary ID • function SLQSGetProfileSettings() returns a default value 0xFF if this parameter is not returned by the device
pIPv6AddPref	• IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network
pUMTSReqQoS-SigInd	• UMTS requested QoS with Signalling Indication flag
pUMTSMinQoS-SigInd	• UMTS minimum QoS with Signalling Indication flag
pPrimaryDNSI-Pv6addpref	• Primary DNS IPv6 address preference – The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP
pSecondaryDN-SIPv6addpref	• Secondary DNS IPv6 address preference
paddrAllocation-Pref	• DHCP/NAS preference – This enumerated value may be used to indicate the address allocation preference * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation
pQosClassID	• 3GPP LTE QoS parameters
pAPNDisabled-Flag	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If set, the profile can not be used for making data calls • Any data call is failed locally • Values: – 0 - FALSE(default) – 1 - True • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().
pPDNInactiv-Timeout	• Optional 4 uint8_ts indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().

pAPNClass	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().

8.53.2 Field Documentation

8.53.2.1 uint8_t* LibpackProfile3GPP::pAddrAllocPref

8.53.2.2 uint8_t* LibpackProfile3GPP::pAPNClass

8.53.2.3 uint8_t* LibpackProfile3GPP::pAPNDisabledFlag

8.53.2.4 uint8_t* LibpackProfile3GPP::pAPNName

8.53.2.5 uint16_t* LibpackProfile3GPP::pAPNnameSize

8.53.2.6 uint8_t* LibpackProfile3GPP::pAuthenticationPref

8.53.2.7 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSMinimumQoS

8.53.2.8 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSRequestedQos

8.53.2.9 uint8_t* LibpackProfile3GPP::pImCnFlag

8.53.2.10 uint32_t* LibpackProfile3GPP::pIPv4AddrPref

8.53.2.11 uint16_t* LibpackProfile3GPP::pIPv6AddPref

8.53.2.12 uint8_t* LibpackProfile3GPP::pPassword

8.53.2.13 uint16_t* LibpackProfile3GPP::pPasswordSize

8.53.2.14 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingDhcp

8.53.2.15 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingPCO

8.53.2.16 uint32_t* LibpackProfile3GPP::pPDNInactivTimeout

8.53.2.17 uint8_t* LibpackProfile3GPP::pPdpAccessConFlag

8.53.2.18 uint8_t* LibpackProfile3GPP::pPdpContext

8.53.2.19 uint8_t* LibpackProfile3GPP::pPdpDataCompType

8.53.2.20 uint8_t* LibpackProfile3GPP::pPdpHdrCompType

8.53.2.21 uint8_t* LibpackProfile3GPP::pPDPtype

8.53.2.22 uint32_t* LibpackProfile3GPP::pPriDNSIPv4AddPref

8.53.2.23 uint16_t* LibpackProfile3GPP::pPriDNSIPv6addpref

8.53.2.24 uint8_t* LibpackProfile3GPP::pPrimaryID

8.53.2.25 uint8_t* LibpackProfile3GPP::pProfilename

8.53.2.26 uint16_t* LibpackProfile3GPP::pProfilenameSize

8.53.2.27 LibPackQosClassID* LibpackProfile3GPP::pQosClassID

8.53.2.28 uint32_t* LibpackProfile3GPP::pSecDNSIPv4AddPref

8.53.2.29 uint16_t* LibpackProfile3GPP::pSecDNSIPv6addpref

8.53.2.30 uint8_t* LibpackProfile3GPP::pSecondaryFlag

8.53.2.31 LibPackTFTIDParams* LibpackProfile3GPP::pTFTID1Params

8.53.2.32 LibPackTFTIDParams* LibpackProfile3GPP::pTFTID2Params

8.53.2.33 LibPackUMTSQoS* LibpackProfile3GPP::pUMTSMinQoS

8.53.2.34 LibPackUMTSReqQoSSigInd* LibpackProfile3GPP::pUMTSMinQosSigInd

8.53.2.35 LibPackUMTSQoS* LibpackProfile3GPP::pUMTSReqQoS

8.53.2.36 LibPackUMTSReqQoSSigInd* LibpackProfile3GPP::pUMTSReqQoSSigInd

8.53.2.37 uint8_t* LibpackProfile3GPP::pUsername

8.53.2.38 uint16_t* LibpackProfile3GPP::pUsernameSize

8.54 LibpackProfile3GPP2 Struct Reference

Data Fields

- uint8_t * pNegoDnsSrvrPref
- uint32_t * pPppSessCloseTimerDO
- uint32_t * pPppSessCloseTimer1x
- uint8_t * pAllowLinger
- uint16_t * pLcpAckTimeout
- uint16_t * pIpcpAckTimeout
- uint16_t * pAuthTimeout
- uint8_t * pLcpCreqRetryCount
- uint8_t * pIpcpCreqRetryCount
- uint8_t * pAuthRetryCount
- uint8_t * pAuthProtocol
- uint8_t * pUserId
- uint16_t * pUserIdSize
- uint8_t * pAuthPassword
- uint16_t * pAuthPasswordSize
- uint8_t * pDataRate
- uint32_t * pAppType
- uint8_t * pDataMode
- uint8_t * pAppPriority
- uint8_t * pApnString
- uint16_t * pApnStringSize

- uint8_t * pPdnType
- uint8_t * pIsPcscfAddressNedded
- uint32_t * pPrimaryV4DnsAddress
- uint32_t * pSecondaryV4DnsAddress
- uint16_t * pPriV6DnsAddress
- uint16_t * pSecV6DnsAddress
- uint8_t * pRATType
- uint8_t * pAPNEnabled3GPP2
- uint32_t * pPDNInactivTimeout3GPP2
- uint8_t * pAPNClass3GPP2

8.54.1 Detailed Description

This structure contains the 3GPP2 profile parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pNegoDnsSrvr-Pref	• Negotiate DNS Server Preference – 1 - (TRUE)implies request DNS addresses from the PDSN – 0 - (FALSE)implies do not request DNS addresses from the PDSN – Default value is 1 (TRUE)
pPppSessClose-TimerDO	• PPP Session Close Timer for DO – Timer value (in seconds) on DO indicating how long the PPP Session should linger before closing down
pPppSessClose-Timer1x	• PPP Session Close Timer for 1X – Timer value (in seconds) on 1X indicating how long the PPP session should linger before closing down
pAllowLinger	• Allow/disallow lingering of interface – 1 -(TRUE) implies allow lingering – 0 -(FALSE) implies do not allow lingering
pLcpAckTimeout	• LCP ACK Timeout – Value of LCP ACK Timeout in milliseconds
pIpcpAck-Timeout	• IPCP ACK Timeout – Value of IPCP ACK Timeout in milliseconds
pAuthTimeout	• AUTH Timeout – Value of Authentication Timeout in milliseconds

pLcpCreqRetry-Count	• LCP Configuration Request Retry Count
pIpcpCreqRetry-Count	• IPCP Configuration Request Retry Count
pAuthRetry-Count	• Authentication Retry Count value
pAuthProtocol	• Authentication Protocol – 1 - PAP – 2 - CHAP – 3 - PAP or CHAP
pUserId	• User ID to be used during data network authentication • maximum length allowed is 127 uint8_ts; • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
pUserIdSize;	• This parameter is an input parameter and should be initialised to the size of pUserId field. Size of this parameter is 2 uint8_ts.
pAuthPassword	• Password to be used during data network authentication; • maximum length allowed is 127 uint8_ts • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
pAuthPassword-Size;	• This parameter is an input parameter and should be initialised to the size of pAuthPassword field. Size of this parameter is 2 uint8_ts.
pDataRate	• Data Rate Requested – 0 - Low (Low speed Service Options (SO15) only) – 1 - Medium (SO33 + low R-SCH) – 2 - High (SO33 + high R-SCH) – Default is 2
pAppType	• Application Type: – 0x00000001 - Default Application Type – 0x00000020 - LBS Application Type – 0x00000040 - Tethered Application Type – This parameter is not used while creating/modifying a profile

pDataMode	• Data Mode to use: – 0 - CDMA or HDR (Hybrid 1X/1xEV-DO) – 1 - CDMA Only (1X only) – 2 - HDR Only (1xEV-DO only) – Default is 0
pAppPriority	• Application Priority – Numerical 1 uint8_t value defining the application priority; higher value implies higher priority – This parameter is not used while creating/modifying a profile
pApnString	• String representing the Access Point Name • maximum length allowed is 100 uint8_ts • QMI_ERR_ARG_TOO_LONG will be returned if the APN name is too long.
pApnStringSize;	• This parameter is an input parameter and should be initialised to the size of pApnString field. Size of this parameter is 2 uint8_ts.
pPdnType	• Packed Data Network Type Requested: – 0 - IPv4 PDN Type – 1 - IPv6 PDN Type – 2 - IPv4 or IPv6 PDN Type – 3 - Unspecified PDN Type (implying no preference)
pIsPcscf-AddressNedded	• This boolean value is used to control if PCSCF address is requested from PDSN – 1 -(TRUE) implies request for PCSCF value from the PDSN – 0 -(FALSE) implies do not request for PCSCF value from the PDSN
pPrimaryV4Dns-Address	• IPv4 Primary DNS address – The Primary IPv4 DNS address that can be statically assigned to the UE
pSecondaryV4-DnsAddress	• IPv4 Secondary DNS address – The Secondary IPv4 DNS address that can be statically assigned to the UE
pPriV6Dns-Address	• Primary IPv6 DNS address – The Primary IPv6 DNS address that can be statically assigned to the UE
pSecV6Dns-Address	• Secondary IPv6 DNS address – The Secondary IPv6 DNS address that can be statically assigned to the UE

pRATType	• Optional 1 uint8_t Flag indicating RAT Type • Values: – 1 - HRPD – 2 - EHRPD – 3 - HRPD_EHRPD • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
pAPNEnabled3GPP2	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If disabled, the profile can not be used for making data calls • Values: – 0 - Disabled – 1 - Enabled(default value) • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
pPDNInactivTimeout3GPP2	• Optional 4 uint8_ts indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
pAPNClass3GPP2	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().

8.54.2 Field Documentation

8.54.2.1 uint8_t* LibpackProfile3GPP2::pAllowLinger

8.54.2.2 uint8_t* LibpackProfile3GPP2::pAPNClass3GPP2

8.54.2.3 uint8_t* LibpackProfile3GPP2::pAPNEnabled3GPP2

8.54.2.4 uint8_t* LibpackProfile3GPP2::pApnString

8.54.2.5 uint16_t* LibpackProfile3GPP2::pApnStringSize

8.54.2.6 uint8_t* LibpackProfile3GPP2::pAppPriority

8.54.2.7 uint32_t* LibpackProfile3GPP2::pAppType

8.54.2.8 uint8_t* LibpackProfile3GPP2::pAuthPassword

8.54.2.9 uint16_t* LibpackProfile3GPP2::pAuthPasswordSize

8.54.2.10 uint8_t* LibpackProfile3GPP2::pAuthProtocol

8.54.2.11 uint8_t* LibpackProfile3GPP2::pAuthRetryCount

8.54.2.12 uint16_t* LibpackProfile3GPP2::pAuthTimeout

8.54.2.13 uint8_t* LibpackProfile3GPP2::pDataMode

8.54.2.14 uint8_t* LibpackProfile3GPP2::pDataRate

8.54.2.15 uint16_t* LibpackProfile3GPP2::pIpcpAckTimeout

8.54.2.16 uint8_t* LibpackProfile3GPP2::pIpcpCreqRetryCount

8.54.2.17 uint8_t* LibpackProfile3GPP2::pIsPcscfAddressNedded

8.54.2.18 uint16_t* LibpackProfile3GPP2::pLcpAckTimeout

8.54.2.19 uint8_t* LibpackProfile3GPP2::pLcpCreqRetryCount

8.54.2.20 uint8_t* LibpackProfile3GPP2::pNegoDnsSrvrPref

8.54.2.21 uint32_t* LibpackProfile3GPP2::pPDNInactivTimeout3GPP2

8.54.2.22 uint8_t* LibpackProfile3GPP2::pPdnType

8.54.2.23 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimer1x

8.54.2.24 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimerDO

8.54.2.25 uint32_t* LibpackProfile3GPP2::pPrimaryV4DnsAddress

8.54.2.26 uint16_t* LibpackProfile3GPP2::pPriV6DnsAddress

8.54.2.27 uint8_t* LibpackProfile3GPP2::pRATType

8.54.2.28 uint32_t* LibpackProfile3GPP2::pSecondaryV4DnsAddress

8.54.2.29 uint16_t* LibpackProfile3GPP2::pSecV6DnsAddress

8.54.2.30 uint8_t* LibpackProfile3GPP2::pUserId

8.54.2.31 uint16_t* LibpackProfile3GPP2::pUserIdSize

8.55 LibPackprofile_3GPP Struct Reference

Data Fields

- uint8_t * pProfilename
- uint16_t * pProfilenameSize
- uint8_t * pPDPtype
- uint8_t * pPdpHdrCompType
- uint8_t * pPdpDataCompType
- uint8_t * pAPNName
- uint16_t * pAPNnameSize
- uint32_t * pPriDNSIPv4AddPref

- uint32_t * pSecDNSIPv4AddPref
- LibPackUMTSQoS * pUMTSReqQoS
- LibPackUMTSQoS * pUMTSMinQoS
- LibPackGPRSRequestedQoS * pGPRSRequestedQos
- LibPackGPRSRequestedQoS * pGPRSMinimumQoS
- uint8_t * pUsername
- uint16_t * pUsernameSize
- uint8_t * pPassword
- uint16_t * pPasswordSize
- uint8_t * pAuthenticationPref
- uint32_t * pIPv4AddrPref
- uint8_t * pPcscfAddrUsingPCO
- uint8_t * pPdpAccessConFlag
- uint8_t * pPcscfAddrUsingDhcp
- uint8_t * pImCnFlag
- LibPackTFTIDParams * pTFTID1Params
- LibPackTFTIDParams * pTFTID2Params
- uint8_t * pPdpContext
- uint8_t * pSecondaryFlag
- uint8_t * pPrimaryID
- uint16_t * pIPv6AddPref
- LibPackUMTSReqQoSSigInd * pUMTSReqQoSSigInd
- LibPackUMTSReqQoSSigInd * pUMTSMinQosSigInd
- uint16_t * pPriDNSIPv6addpref
- uint16_t * pSecDNSIPv6addpref
- uint8_t * pAddrAllocPref
- LibPackQosClassID * pQosClassID
- uint8_t * pAPNDisabledFlag
- uint32_t * pPDNInactivTimeout
- uint8_t * pAPNClass

8.55.1 Detailed Description

This structure contains Input/Output parameters of pack_wds_SLQSCreateProfile

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pProfileName	• One or more bytes describing the profile
pProfilename-Size;	• This parameter is an input parameter and should be initialised to the size of pProfileName field. Size of this parameter is 2 bytes.
pPDPType	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0x00 - PDP-IP (IPv4) – 0x01 - PDP-PPP – 0x02 - PDP-IPV6 – 0x03 - PDP-IPV4V6

pPdpHdrComp-Type	• PDP header compression type – 0 - PDP header compression is OFF – 1 - Manufacturer preferred compression – 2 - PDP header compression based on RFC 1144 – 3 - PDP header compression based on RFC 25074 PDP header compression based on RFC 3095
pPdpDataComp-Type	• PDP data compression type – 0 - PDP data compression is OFF – 1 - Manufacturer preferred compression – 2 - V.42BIS data compression – 3 - V.44 data compression
pAPNName	• Access point name
pAPNnameSize;	• This parameter is an input parameter and should be initialised to the size of pAPNName field. Size of this parameter is 2 bytes.
pPriDNSIPv4-AddPref	• Primary DNS IPv4 Address Preference
pSecDNSIPv4-AddPref	• Secondary DNS IPv4 Address Preference
pUMTSReqQoS	• UMTS Requested QoS
pUMTSMinQoS	• UMTS Minimum QoS
pGPRS-RequestedQoS	• GPRS Minimum QoS
pUsername	• User name
pUsernameSize;	• This parameter is an input parameter and should be initialised to the size of pUsername field. Size of this parameter is 2 bytes.
pPassword	• Password
pPasswordSize;	• This parameter is an input parameter and should be initialised to the size of pPassword field. Size of this parameter is 2 bytes.

pAuthentication-Pref	• Authentication Preference – Bit map that indicates the authentication algorithm preference * Bit 0 - PAP preference · 0 - PAP is never performed · 1 - PAP may be performed * Bit 1 - CHAP preference · 0 - CHAP is never performed · 1 - CHAP may be performed * If more than one bit is set, then the device decides which authentication procedure is performed while setting up the data session. For example, the device may have a policy to select the most secure authentication mechanism.
pIPv4AddrPref	• IPv4 Address Preference
pPcscfAddr-UsingPCO	• P-CSCF Address using PCO Flag – 1 - (TRUE) implies request PCSCF address using PCO – 0 - (FALSE) implies do not request By default, this value is 0
pPdpAccess-ConFlag	• PDP access control flag – 0 - PDP access control none – 1 - PDP access control reject – 2 - PDP access control permission
pPcscfAddr-UsingDhcp	• P-CSCF address using DHCP – 1 - (TRUE) implies Request PCSCF address using DHCP – 0 - (FALSE) implies do not request By default, value is 0
pImCnFlag	• IM CN flag – 1 - (TRUE) implies request IM CN flag for this profile – 0 - (FALSE) implies do not request IM CN flag for this profile
pTFTID1Params	• Traffic Flow Template
pTFTID2Params	• Traffic Flow Template
pPdpContext	• PDP context number
pSecondaryFlag	• PDP context secondary flag – 1 - (TRUE) implies this is secondary profile – 0 - (FALSE) implies this is not secondary profile

pPrimaryID	• PDP context primary ID • function SLQSGetProfileSettings() returns a default value 0xFF if this parameter is not returned by the device
pIPv6AddPref	• IPv6 address preference Preferred IPv6 address to be assigned to the TE; actual assigned address is negotiated with the network and may differ from this value; if not specified, the IPv6 address is obtained automatically from the network
pUMTSReqQoS-SigInd	• UMTS requested QoS with Signalling Indication flag
pUMTSMinQoS-SigInd	• UMTS minimum QoS with Signalling Indication flag
pPrimaryDNSI-Pv6addpref	• Primary DNS IPv6 address preference – The value may be used as a preference during negotiation with the network; if not specified, the wireless device will attempt to obtain the DNS address automatically from the network; the negotiated value is provided to the host via DHCP
pSecondaryDN-SIPv6addpref	• Secondary DNS IPv6 address preference
paddrAllocation-Pref	• DHCP/NAS preference – This enumerated value may be used to indicate the address allocation preference * 0 - NAS signaling is used for address allocation * 1 - DHCP is used for address allocation
pQosClassID	• 3GPP LTE QoS parameters
pAPNDisabled-Flag	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If set, the profile can not be used for making data calls • Any data call is failed locally • Values: – 0 - FALSE(default) – 1 - True • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().
pPDNInactiv-Timeout	• Optional 4 Bytes indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().

pAPNClass	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfile-Settings().

8.55.2 Field Documentation

8.55.2.1 uint8_t* LibPackprofile_3GPP::pAddrAllocPref

8.55.2.2 uint8_t* LibPackprofile_3GPP::pAPNClass

8.55.2.3 uint8_t* LibPackprofile_3GPP::pAPNDisabledFlag

8.55.2.4 uint8_t* LibPackprofile_3GPP::pAPNName

8.55.2.5 uint16_t* LibPackprofile_3GPP::pAPNnameSize

8.55.2.6 uint8_t* LibPackprofile_3GPP::pAuthenticationPref

8.55.2.7 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSMinimumQoS

8.55.2.8 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSRequestedQos

8.55.2.9 uint8_t* LibPackprofile_3GPP::pImCnFlag

8.55.2.10 uint32_t* LibPackprofile_3GPP::pIPv4AddrPref

8.55.2.11 uint16_t* LibPackprofile_3GPP::pIPv6AddPref

8.55.2.12 uint8_t* LibPackprofile_3GPP::pPassword

8.55.2.13 uint16_t* LibPackprofile_3GPP::pPasswordSize

8.55.2.14 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingDhcp

8.55.2.15 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingPCO

8.55.2.16 uint32_t* LibPackprofile_3GPP::pPDNInactivTimeout

8.55.2.17 uint8_t* LibPackprofile_3GPP::pPdpAccessConFlag

8.55.2.18 uint8_t* LibPackprofile_3GPP::pPdpContext

8.55.2.19 uint8_t* LibPackprofile_3GPP::pPdpDataCompType

8.55.2.20 uint8_t* LibPackprofile_3GPP::pPdpHdrCompType

8.55.2.21 uint8_t* LibPackprofile_3GPP::pPDPtype

8.55.2.22 uint32_t* LibPackprofile_3GPP::pPriDNSIPv4AddPref

8.55.2.23 uint16_t* LibPackprofile_3GPP::pPriDNSIPv6addpref

8.55.2.24 uint8_t* LibPackprofile_3GPP::pPrimaryID

8.55.2.25 uint8_t* LibPackprofile_3GPP::pProfilename

8.55.2.26 uint16_t* LibPackprofile_3GPP::pProfilenameSize

8.55.2.27 LibPackQosClassID* LibPackprofile_3GPP::pQosClassID

8.55.2.28 uint32_t* LibPackprofile_3GPP::pSecDNSIPv4AddPref

8.55.2.29 uint16_t* LibPackprofile_3GPP::pSecDNSIPv6addpref

8.55.2.30 uint8_t* LibPackprofile_3GPP::pSecondaryFlag

8.55.2.31 LibPackTFTIDParams* LibPackprofile_3GPP::pTFTID1Params

8.55.2.32 LibPackTFTIDParams* LibPackprofile_3GPP::pTFTID2Params

8.55.2.33 LibPackUMTSQoS* LibPackprofile_3GPP::pUMTSMinQoS

8.55.2.34 LibPackUMTSReqQoSSigInd* LibPackprofile_3GPP::pUMTSMinQosSigInd

8.55.2.35 LibPackUMTSQoS* LibPackprofile_3GPP::pUMTSReqQoS

8.55.2.36 LibPackUMTSReqQoSSigInd* LibPackprofile_3GPP::pUMTSReqQoSSigInd

8.55.2.37 uint8_t* LibPackprofile_3GPP::pUsername

8.55.2.38 uint16_t* LibPackprofile_3GPP::pUsernameSize

8.56 LibPackprofile_3GPP2 Struct Reference

Data Fields

- uint8_t * pNegoDnsSrvrPref
- uint32_t * pPppSessCloseTimerDO
- uint32_t * pPppSessCloseTimer1x
- uint8_t * pAllowLinger
- uint16_t * pLcpAckTimeout
- uint16_t * pIpcpAckTimeout
- uint16_t * pAuthTimeout
- uint8_t * pLcpCreqRetryCount
- uint8_t * pIpcpCreqRetryCount
- uint8_t * pAuthRetryCount
- uint8_t * pAuthProtocol
- uint8_t * pUserId
- uint16_t * pUserIdSize
- uint8_t * pAuthPassword
- uint16_t * pAuthPassword_tSize
- uint8_t * pDataRate
- uint32_t * pAppType
- uint8_t * pDataMode
- uint8_t * pAppPriority
- uint8_t * pApnString
- uint16_t * pApnStringSize

- uint8_t * pPdnType
- uint8_t * pIsPcscfAddressNedded
- uint32_t * pPrimaryV4DnsAddress
- uint32_t * pSecondaryV4DnsAddress
- uint16_t * pPriV6DnsAddress
- uint16_t * pSecV6DnsAddress
- uint8_t * pRATType
- uint8_t * pAPNEnabled3GPP2
- uint32_t * pPDNInactivTimeout3GPP2
- uint8_t * pAPNClass3GPP2

8.56.1 Detailed Description

This structure contains the 3GPP2 profile parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

pNegoDnsSrvr-Pref	• Negotiate DNS Server Preference – 1 - (TRUE)implies request DNS addresses from the PDSN – 0 - (FALSE)implies do not request DNS addresses from the PDSN – Default value is 1 (TRUE)
pPppSessClose-TimerDO	• PPP Session Close Timer for DO – Timer value (in seconds) on DO indicating how long the PPP Session should linger before closing down
pPppSessClose-Timer1x	• PPP Session Close Timer for 1X – Timer value (in seconds) on 1X indicating how long the PPP session should linger before closing down
pAllowLinger	• Allow/disallow lingering of interface – 1 -(TRUE) implies allow lingering – 0 -(FALSE) implies do not allow lingering
pLcpAckTimeout	• LCP ACK Timeout – Value of LCP ACK Timeout in milliseconds
pIpcpAck-Timeout	• IPCP ACK Timeout – Value of IPCP ACK Timeout in milliseconds
pAuthTimeout	• AUTH Timeout – Value of Authentication Timeout in milliseconds

pLcpCreqRetry-Count	• LCP Configuration Request Retry Count
pIpcpCreqRetry-Count	• IPCP Configuration Request Retry Count
pAuthRetry-Count	• Authentication Retry Count value
pAuthProtocol	• Authentication Protocol – 1 - PAP – 2 - CHAP – 3 - PAP or CHAP
pUserId	• User ID to be used during data network authentication • maximum length allowed is 127 bytes; • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
pUserIdSize;	• This parameter is an input parameter and should be initialised to the size of pUserId field. Size of this parameter is 2 bytes.
pAuthPassword	• Password to be used during data network authentication; • maximum length allowed is 127 bytes • QMI_ERR_ARG_TOO_LONG will be returned if the storage on the wireless device is insufficient in size to hold the value.
pAuthPassword-Size;	• This parameter is an input parameter and should be initialised to the size of pAuthPassword field. Size of this parameter is 2 bytes.
pDataRate	• Data Rate Requested – 0 - Low (Low speed Service Options (SO15) only) – 1 - Medium (SO33 + low R-SCH) – 2 - High (SO33 + high R-SCH) – Default is 2
pAppType	• Application Type: – 0x00000001 - Default Application Type – 0x00000020 - LBS Application Type – 0x00000040 - Tethered Application Type – This parameter is not used while creating/modifying a profile

pDataMode	• Data Mode to use: – 0 - CDMA or HDR (Hybrid 1X/1xEV-DO) – 1 - CDMA Only (1X only) – 2 - HDR Only (1xEV-DO only) – Default is 0
pAppPriority	• Application Priority – Numerical 1 uint8_t value defining the application priority; higher value implies higher priority – This parameter is not used while creating/modifying a profile
pApnString	• String representing the Access Point Name • maximum length allowed is 100 bytes • QMI_ERR_ARG_TOO_LONG will be returned if the APN name is too long.
pApnStringSize;	• This parameter is an input parameter and should be initialised to the size of pApnString field. Size of this parameter is 2 bytes.
pPdnType	• Packed Data Network Type Requested: – 0 - IPv4 PDN Type – 1 - IPv6 PDN Type – 2 - IPv4 or IPv6 PDN Type – 3 - Unspecified PDN Type (implying no preference)
pIsPcscf-AddressNedded	• This boolean value is used to control if PCSCF address is requested from PDSN – 1 -(TRUE) implies request for PCSCF value from the PDSN – 0 -(FALSE) implies do not request for PCSCF value from the PDSN
pPrimaryV4Dns-Address	• IPv4 Primary DNS address – The Primary IPv4 DNS address that can be statically assigned to the UE
pSecondaryV4-DnsAddress	• IPv4 Secondary DNS address – The Secondary IPv4 DNS address that can be statically assigned to the UE
pPriV6Dns-Address	• Primary IPv6 DNS address – The Primary IPv6 DNS address that can be statically assigned to the UE
pSecV6Dns-Address	• Secondary IPv6 DNS address – The Secondary IPv6 DNS address that can be statically assigned to the UE

pRATType	• Optional 1 uint8_t Flag indicating RAT Type • Values: – 1 - HRPD – 2 - EHRPD – 3 - HRPD_EHRPD • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
pAPNEnabled3GPP2	• Optional 1 uint8_t Flag indicating if the APN is disabled/enabled • If disabled, the profile can not be used for making data calls • Values: – 0 - Disabled – 1 - Enabled(default value) • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
pPDNInactivTimeout3GPP2	• Optional 4 Bytes indicating the duration of inactivity timer in seconds • If the PDP context/PDN connection is inactive for this duration i.e. No data Tx/Rx occurs, the PDP context/PDN connection is disconnected • Default value of zero indicates infinite value • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().
pAPNClass3GPP2	• Optional 1 uint8_t numeric identifier representing the APN in profile • Can be set and queried but is not used by the modem • This parameter is currently read only and can be read by using the function SLQSGetProfileSettings().

8.56.2 Field Documentation

8.56.2.1 uint8_t* LibPackprofile_3GPP2::pAllowLinger

8.56.2.2 uint8_t* LibPackprofile_3GPP2::pAPNClass3GPP2

8.56.2.3 uint8_t* LibPackprofile_3GPP2::pAPNEnabled3GPP2

8.56.2.4 uint8_t* LibPackprofile_3GPP2::pApnString

8.56.2.5 uint16_t* LibPackprofile_3GPP2::pApnStringSize

8.56.2.6 uint8_t* LibPackprofile_3GPP2::pAppPriority

8.56.2.7 uint32_t* LibPackprofile_3GPP2::pAppType

8.56.2.8 uint8_t* LibPackprofile_3GPP2::pAuthPassword

8.56.2.9 uint16_t* LibPackprofile_3GPP2::pAuthPassword_tSize

8.56.2.10 uint8_t* LibPackprofile_3GPP2::pAuthProtocol

8.56.2.11 uint8_t* LibPackprofile_3GPP2::pAuthRetryCount

8.56.2.12 uint16_t* LibPackprofile_3GPP2::pAuthTimeout

8.56.2.13 uint8_t* LibPackprofile_3GPP2::pDataMode

8.56.2.14 uint8_t* LibPackprofile_3GPP2::pDataRate

8.56.2.15 uint16_t* LibPackprofile_3GPP2::pIpcpAckTimeout

8.56.2.16 uint8_t* LibPackprofile_3GPP2::pIpcpCreqRetryCount

8.56.2.17 uint8_t* LibPackprofile_3GPP2::pIsPcscfAddressNedded

8.56.2.18 uint16_t* LibPackprofile_3GPP2::pLcpAckTimeout

8.56.2.19 uint8_t* LibPackprofile_3GPP2::pLcpCreqRetryCount

8.56.2.20 uint8_t* LibPackprofile_3GPP2::pNegoDnsSrvrPref

8.56.2.21 uint32_t* LibPackprofile_3GPP2::pPDNInactivTimeout3GPP2

8.56.2.22 uint8_t* LibPackprofile_3GPP2::pPdnType

8.56.2.23 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimer1x

8.56.2.24 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimerDO

8.56.2.25 uint32_t* LibPackprofile_3GPP2::pPrimaryV4DnsAddress

8.56.2.26 uint16_t* LibPackprofile_3GPP2::pPriV6DnsAddress

8.56.2.27 uint8_t* LibPackprofile_3GPP2::pRATType

8.56.2.28 uint32_t* LibPackprofile_3GPP2::pSecondaryV4DnsAddress

8.56.2.29 uint16_t* LibPackprofile_3GPP2::pSecV6DnsAddress

8.56.2.30 uint8_t* LibPackprofile_3GPP2::pUserId

8.56.2.31 uint16_t* LibPackprofile_3GPP2::pUserIdSize

8.57 LibPackQosClassID Struct Reference

Data Fields

- uint8_t QCI
- uint8_t gDlBitRate
- uint32_t maxDlBitRate
- uint32_t gUlBitRate
- uint32_t maxUlBitRate

8.57.1 Detailed Description

structure contains 3GPP LTE QoS parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

QCI	• QOS specified using the QOS Class Identifier (QOS) values QCI value 0 - Requests the network to assign the appropriate QCI value QCI values 1-4 - Associated with guaranteed bit rates QCI values 5-9 - Associated with non-guaranteed bit rates
gDlBitRate	• Guaranteed DL bit rate
maxDlBitRate	• Maximum DL bit rate
gUlBitRate	• Guaranteed UL bit rate
maxUlBitRate	• Maximum UL bit rate

8.57.2 Field Documentation

8.57.2.1 uint8_t LibPackQosClassID::gDlBitRate

8.57.2.2 uint32_t LibPackQosClassID::gUlBitRate

8.57.2.3 uint32_t LibPackQosClassID::maxDlBitRate

8.57.2.4 uint32_t LibPackQosClassID::maxUlBitRate

8.57.2.5 uint8_t LibPackQosClassID::QCI

8.58 LibPackTFTIDParams Struct Reference

Data Fields

- uint8_t filterId
- uint8_t eValid
- uint8_t ipVersion
- uint16_t * pSourceIP
- uint8_t sourceIPMask
- uint8_t nextHeader
- uint32_t destPortRangeStart
- uint16_t destPortRangeEnd
- uint16_t srcPortRangeStart
- uint16_t srcPortRangeEnd
- uint32_t IPSECSPI
- uint16_t tosMask
- uint32_t flowLabel

8.58.1 Detailed Description

structure contains traffic flow template parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

filterId	• Filter identifier
eValid	• Evaluation precedence index
ipVersion	• IP version number – 4 - IPv4 – 6 - IPv6
pSourceIP	• Source IP address – IPv4 - Fill the first 4 uint8_ts – IPv6 - Fill all the 16 uint8_ts
sourceIPMask	• Mask value for the source address
nextHeader	• Next header/protocol value
destPortRange-Start	• Start value of the destination port range
destPortRange-End	• End value of the destination port range
srcPortRange-Start	• Start value of the source port range
srcPortRange-End	• End value of the source port range
IPSECSPI	• IPSEC security parameter index
tosMask	• TOS mask (Traffic class for IPv6)
flowLabel	• Flow label

8.58.2 Field Documentation

8.58.2.1 uint16_t LibPackTFTIDParams::destPortRangeEnd

8.58.2.2 uint32_t LibPackTFTIDParams::destPortRangeStart

8.58.2.3 uint8_t LibPackTFTIDParams::eValid

8.58.2.4 uint8_t LibPackTFTIDParams::filterId

8.58.2.5 uint32_t LibPackTFTIDParams::flowLabel

8.58.2.6 uint32_t LibPackTFTIDParams::IPSECSPI

8.58.2.7 uint8_t LibPackTFTIDParams::ipVersion

8.58.2.8 uint8_t LibPackTFTIDParams::nextHeader

8.58.2.9 uint16_t* LibPackTFTIDParams::pSourceIP

8.58.2.10 uint8_t LibPackTFTIDParams::sourceIPMask

8.58.2.11 uint16_t LibPackTFTIDParams::srcPortRangeEnd

8.58.2.12 uint16_t LibPackTFTIDParams::srcPortRangeStart

8.58.2.13 uint16_t LibPackTFTIDParams::tosMask

8.59 LibPackUMTSQoS Struct Reference

Data Fields

- uint8_t trafficClass
- uint32_t maxUplinkBitrate
- uint32_t maxDownlinkBitrate
- uint32_t grntUplinkBitrate
- uint32_t grntDownlinkBitrate
- uint8_t qosDeliveryOrder
- uint32_t maxSDUSize
- uint8_t sduErrorRatio
- uint8_t resBerRatio
- uint8_t deliveryErrSDU
- uint32_t transferDelay
- uint32_t trafficPriority

8.59.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

trafficClass	• 0x00 - Subscribed • 0x01 - Conversational • 0x02 - Streaming • 0x03 - Interactive • 0x04 - Background
maxUplinkBitrate	• Maximum uplink bit rate in bits/sec
maxDownlink-Bitrate	• Maximum downlink bit rate in bits/sec
grntUplinkBitrate	• Guaranteed uplink bit rate in bits/sec
grntDownlink-Bitrate	• Guranteed downlink bit rate in bits/sec
qosDelivery-Order	- Qos delivery order • 0x00 - Subscribe • 0x01 - delivery order on • 0x02 - delivery order off
maxSDUSize	• Maximum SDU size
sduErrorRatio	- SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - 1∗10^(-2) • 0x02 - 7∗10^(-3) • 0x03 - 1∗10^(-3) • 0x04 - 1∗10^(-4) • 0x05 - 1∗10^(-5) • 0x06 - 1∗10^(-6) • 0x07 - 1∗10^(-1)
resBerRatio	- Residual bit error ratio • Target value for undetected bit error ratio in in the delivered SDUs. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)

deliveryErrSDU	- Delivery of erroneous SDUs • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)
transferDelay	- Transfer delay (ms) • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
trafficPriority	- Transfer handling priority • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

8.59.2 Field Documentation

8.59.2.1 uint8_t LibPackUMTSQoS::deliveryErrSDU

8.59.2.2 uint32_t LibPackUMTSQoS::grntDownlinkBitrate

8.59.2.3 uint32_t LibPackUMTSQoS::grntUplinkBitrate

8.59.2.4 uint32_t LibPackUMTSQoS::maxDownlinkBitrate

8.59.2.5 uint32_t LibPackUMTSQoS::maxSDUSize

8.59.2.6 uint32_t LibPackUMTSQoS::maxUplinkBitrate

8.59.2.7 uint8_t LibPackUMTSQoS::qosDeliveryOrder

8.59.2.8 uint8_t LibPackUMTSQoS::resBerRatio

8.59.2.9 uint8_t LibPackUMTSQoS::sduErrorRatio

8.59.2.10 uint8_t LibPackUMTSQoS::trafficClass

8.59.2.11 uint32_t LibPackUMTSQoS::trafficPriority

8.59.2.12 uint32_t LibPackUMTSQoS::transferDelay

8.60 LibPackUMTSReqQoSSigInd Struct Reference

Data Fields

- LibPackUMTSQoS UMTSReqQoS
- uint8_t SigInd

8.60.1 Detailed Description

structure contains UMTS requested QoS with Signaling Indication flag

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

UMTSReqQoS	• Contains the UMTS Quality Of Service Information
SigInd	- Signaling Indication flag • TRUE - Signaling indication ON • FALSE - Signaling indication OFF

8.60.2 Field Documentation

8.60.2.1 uint8_t LibPackUMTSReqQoSSigInd::SigInd

8.60.2.2 LibPackUMTSQoS LibPackUMTSReqQoSSigInd::UMTSReqQoS

8.61 loc_BdsSV Struct Reference

Data Fields

- uint16_t id
- uint8_t mask

8.61.1 Detailed Description

This structure contains the BDS SV Info

Parameters

id	• SV ID of the satellite whose data is to be deleted. – Range for BDS: 201 to 237
mask	• Indicates if the ephemeris or almanac for a satellite is to be deleted • Valid values: – QMI_LOC_MASK_DELETE_EPHEMERIS (0x01) - Delete ephemeris for the satellite – QMI_LOC_MASK_DELETE_ALMANAC (0x02) - Delete almanac for the satellite

8.61.2 Field Documentation

8.61.2.1 uint16_t loc_BdsSV::id

8.61.2.2 uint8_t loc_BdsSV::mask

8.62 loc_BdsSVInfo Struct Reference

Data Fields

- uint8_t len
- loc_BdsSV * pSV

8.62.1 Detailed Description

This structure contains the number of sets of the BDS SVN Info

Parameters

len	• Number of sets of the following elements: – gnssSvId – deleteSvInfoMask
pSV	• Pointer to struct loc_BdsSV. See loc_BdsSV for more information

8.62.2 Field Documentation

8.62.2.1 uint8_t loc_BdsSVInfo::len

8.62.2.2 loc_BdsSV* loc_BdsSVInfo::pSV

8.63 loc_CellDb Struct Reference

Data Fields

- uint32_t mask

8.63.1 Detailed Description

This structure contains the cell database

Parameters

mask	• Mask for the cell database assistance data that is to be deleted • Valid values: – 0x00000001 - DELETE_CELLDB_POS – 0x00000002 - DELETE_CELLDB_LATEST_GPS_POS – 0x00000004 - DELETE_CELLDB_OTA_POS – 0x00000008 - DELETE_CELLDB_EXT_REF_POS – 0x00000010 - DELETE_CELLDB_TIMETAG – 0x00000020 - DELETE_CELLDB_CELLID – 0x00000040 - DELETE_CELLDB_CACHED_CELLID – 0x00000080 - DELETE_CELLDB_LAST_SRV_CELL – 0x00000100 - DELETE_CELLDB_CUR_SRV_CELL – 0x00000200 - DELETE_CELLDB_NEIGHBOR_INFO

8.63.2 Field Documentation

8.63.2.1 uint32_t loc_CellDb::mask

8.64 loc_ClkInfo Struct Reference

Data Fields

- uint32_t mask

8.64.1 Detailed Description

This structure contains the clock Info

Parameters

mask	• Mask for the clock information assistance data that is to be deleted • Valid bitmasks: – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_EST (0x00000001) - Mask to delete time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_FREQ_EST (0x00000002) - Mask to delete frequency estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_WEEK_NUMBER (0x00000004) - Mask to delete week number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_RTC_TIME (0x00000008) - Mask to delete RTC time from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_TIME_TRANSFER (0x00000010) - Mask to delete time transfer from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GPSTIME_EST (0x00000020) - Mask to delete GPS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLOTIME_EST (0x00000040) - Mask to delete GLONASS time estimate from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLODAY_NUMBER (0x00000080) - Mask to delete GLONASS day number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO4YEAR_NUMBER (0x00000100) - Mask to delete GLONASS four year number from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GLO_RF_GRP_DELAY (0x00000200) - Mask to delete GLONASS RF GRP delay from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_DISABLE_TT (0x00000400) - Mask to delete disable TT from clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_LEAPSEC (0x00000800) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GG_GGTB (0x00001000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSTIME_EST (0x00002000) - Mask to delete a BDS time estimate from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_GB_GBTB (0x00004000) - Mask to delete Glonass-to-BDS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BG_BGTB (0x00008000) - Mask to delete BDS-to-GLONASS time bias-related information from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDSWEEK_NUMBER (0x00010000) - Mask to delete the BDS week number from the clock information – QMI_LOC_MASK_DELETE_CLOCK_INFO_BDS_RF_GRP_DELAY (0x00020000) - Mask to delete the BDS RF GRP delay from the clock information

8.64.2 Field Documentation

8.64.2.1 uint32_t loc_ClkInfo::mask

8.65 loc_GnssData Struct Reference

Data Fields

- uint64_t mask

8.65.1 Detailed Description

This structure contains the GNSS data

Parameters

mask	• Mask for the GNSS data that is to be deleted • Valid values: – QMI_LOC_MASK_DELETE_GPS_SVDIR (0x00000001) - Mask to delete GPS SVDIR – QMI_LOC_MASK_DELETE_GPS_SVSTEER (0x00000002) - Mask to delete GPS SVSTEER – QMI_LOC_MASK_DELETE_GPS_TIME (0x00000004) - Mask to delete GPS time – QMI_LOC_MASK_DELETE_GPS_ALM_CORR (0x00000008) - Mask to delete almanac correlation – QMI_LOC_MASK_DELETE_GLO_SVDIR (0x00000010) - Mask to delete GLONASS SVDIR – QMI_LOC_MASK_DELETE_GLO_SVSTEER (0x00000020) - Mask to delete GLONASS SVSTEER – QMI_LOC_MASK_DELETE_GLO_TIME (0x00000040) - Mask to delete GLONASS time – QMI_LOC_MASK_DELETE_GLO_ALM_CORR (0x00000080) - Mask to delete GLONASS almanac correlation – QMI_LOC_MASK_DELETE_SBAS_SVDIR (0x00000100) - Mask to delete SBAS SVDIR – QMI_LOC_MASK_DELETE_SBAS_SVSTEER (0x00000200) - Mask to delete SBAS SVSTEER – QMI_LOC_MASK_DELETE_POSITION (0x00000400) - Mask to delete position estimate – QMI_LOC_MASK_DELETE_TIME (0x00000800) - Mask to delete time estimate – QMI_LOC_MASK_DELETE_IONO (0x00001000) - Mask to delete IONO – QMI_LOC_MASK_DELETE_UTC (0x00002000) - Mask to delete UTC estimate – QMI_LOC_MASK_DELETE_HEALTH (0x00004000) - Mask to delete SV health record – QMI_LOC_MASK_DELETE_SADATA (0x00008000) - Mask to delete SADATA – QMI_LOC_MASK_DELETE_RTI (0x00010000) - Mask to delete RTI – QMI_LOC_MASK_DELETE_SV_NO_EXIST (0x00020000) - Mask to delete SV_NO_EXIST – QMI_LOC_MASK_DELETE_FREQ_BIAS_EST (0x00040000) - Mask to delete frequency bias estimate – QMI_LOC_MASK_DELETE_BDS_SVDIR (0x00080000) - Mask to delete BDS SVDIR – QMI_LOC_MASK_DELETE_BDS_SVSTEER (0x00100000) - Mask to delete BDS SVSTEER – QMI_LOC_MASK_DELETE_BDS_TIME (0x00200000) - Mask to delete BDS time – QMI_LOC_MASK_DELETE_BDS_ALM_CORR (0x00400000) - Mask to delete BDS almanac correlation – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GPS (0x00800000) - Mask to delete GNSS SV blacklist GPS – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_GLO (0x01000000) - Mask to delete GNSS SV blacklist GLO – QMI_LOC_MASK_DELETE_GNSS_SV_BLACKLIST_BDS (0x02000000) - Mask to delete GNSS SV blacklist BDS

8.65.2 Field Documentation

8.65.2.1 uint64_t loc_GnssData::mask

8.66 loc_gpsTime Struct Reference

Data Fields

- uint16_t gpsWeek
- uint32_t gpsTimeOfWeekMs

8.66.1 Detailed Description

This structure contains GPS Time info.

Parameters

gpsWeek	• Current GPS week as calculated from midnight, Jan. 6, 1980. • Units - Weeks
gpsTimeOf-WeekMs	• Amount of time into the current GPS week. • Units - Milliseconds

8.66.2 Field Documentation

8.66.2.1 uint32_t loc_gpsTime::gpsTimeOfWeekMs

8.66.2.2 uint16_t loc_gpsTime::gpsWeek

8.67 loc_IPv4Info Struct Reference

Data Fields

- uint32_t address
- uint16_t port
- uint8_t TlvPresent

8.67.1 Detailed Description

This structure contains LOC Server IPV4 info field.

Parameters

address	• IPv4 address.
port	• IPv4 port.
TlvPresent	• Tlv Present or not

8.67.2 Field Documentation

8.67.2.1 uint32_t loc_IPv4Info::address

8.67.2.2 uint16_t loc_IPv4Info::port

8.67.2.3 uint8_t loc_IPv4Info::TlvPresent

8.68 loc_IPv6Info Struct Reference

Data Fields

- uint16_t address [8]
- uint32_t port
- uint8_t TlvPresent

8.68.1 Detailed Description

This structure contains LOC Get Server IPV6 info field.

Parameters

address	• IPv6 address. • Type - Array of unsigned integers • Maximum length of the array - 8
port	• IPv6 port.
TlvPresent	• Tlv Present or not

8.68.2 Field Documentation

8.68.2.1 uint16_t loc_IPv6Info::address[8]

8.68.2.2 uint32_t loc_IPv6Info::port

8.68.2.3 uint8_t loc_IPv6Info::TlvPresent

8.69 loc_LocApplicationInfo Struct Reference

Data Fields

- uint8_t appProviderLength
- uint8_t * pAppProvider
- uint8_t appNameLength
- uint8_t * pAppName
- uint8_t appVersionValid
- uint8_t appVersionLength
- uint8_t * pAppVersion

8.69.1 Detailed Description

This structure contains the Application Information

Parameters

appProvider-Length	• Length of the Application Provider
pAppProvider	• Application Provider • Depends upon the Length of application Provider
appNameLength	• Length of Application Name
pAppName	• Application Name • Depends upon the Length of application Name
appVersionValid	• Specifies whether the application version string contains a valid value • 0x00 (FALSE) Application version string is invalid • 0x01 (TRUE) Application version string is valid
appVersion-Length	• Length of Application Version
pAppVersion	• Application Version • Depends upon the Length of application Version

8.69.2 Field Documentation

8.69.2.1 uint8_t loc_LocApplicationInfo::appNameLength

8.69.2.2 uint8_t loc_LocApplicationInfo::appProviderLength

8.69.2.3 uint8_t loc_LocApplicationInfo::appVersionLength

8.69.2.4 uint8_t loc_LocApplicationInfo::appVersionValid

8.69.2.5 uint8_t* loc_LocApplicationInfo::pAppName

8.69.2.6 uint8_t* loc_LocApplicationInfo::pAppProvider

8.69.2.7 uint8_t* loc_LocApplicationInfo::pAppVersion

8.70 loc_precisionDilution Struct Reference

Data Fields

- uint32_t PDOP

- uint32_t HDOP
- uint32_t VDOP

8.70.1 Detailed Description

This structure contains Dilution of precision associated with this position.

Parameters

PDOP	• Position dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy) • PDOP = square root of (Square of HDOP + Square of VDOP2)
HDOP	• Horizontal dilution of precision. • Range - 1 (highest accuracy) to 50 (lowest accuracy)
VDOP	• Vertical dilution of precision. • Range- 1 (highest accuracy) to 50 (lowest accuracy)

8.70.2 Field Documentation

8.70.2.1 uint32_t loc_precisionDilution::HDOP

8.70.2.2 uint32_t loc_precisionDilution::PDOP

8.70.2.3 uint32_t loc_precisionDilution::VDOP

8.71 loc_satelliteInfo Struct Reference

Data Fields

- uint8_t svListLen
- uint32_t validMask
- uint32_t system
- uint16_t gnssSvId
- uint8_t healthStatus
- uint32_t svStatus
- uint8_t svInfoMask
- float elevation
- float azimuth
- float snr

8.71.1 Detailed Description

Contain fields in struct loc_satelliteInfo

Parameters

svListLen	• number of sets of the following elements: – validMask – system – gnssSvid – healthStatus – svStatus – svInfoMask – elevation – azimuth – snr
validMask	• Bitmask indicating which of the fields in this TLV are valid. Valid bitmasks: – 0x00000001 - VALID_SYSTEM – 0x00000002 - VALID_GNSS_SVID – 0x00000004 - VALID_HEALTH_STATUS – 0x00000008 - VALID_PROCESS_STATUS – 0x00000010 - VALID_SVINFO_MASK – 0x00000020 - VALID_ELEVATION – 0x00000040 - VALID_AZIMUTH – 0x00000080 - VALID_SNR
system	• Indicates to which constellation this SV belongs. Valid values: – eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite – eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite – eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite – eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite – eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite – eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite
gnssSvId	• GNSS SV ID. The GPS and GLONASS SVs can be disambiguated using the system field. Range: – FOR GPS: 1 to 32 – FOR GLONASS: 1 to 32 – FOR SBAS: 120 to 151 – for BDS: 201 to 237
healthStatus	• health status. Range: 0 - 1 – 0 - unhealthy – 1 - healthy
svStatus	• SV process status. Valid values: – eQMI_LOC_SV_STATUS_IDLE (1) - SV is not being actively processed – eQMI_LOC_SV_STATUS_SEARCH (2) - The system is searching for this SV – eQMI_LOC_SV_STATUS_TRACK (3) - SV is being tracked

svInfoMask	• Indicates whether almanac and ephemeris information is available. Valid bitmasks: – 0x01 - SVINFO_HAS_EPHEMERIS – 0x02 - SVINFO_HAS_ALMANAC
elevation	• SV elevation angle. – Units: Degrees – Range: 0 to 90
azimuth	• SV azimuth angle. – Units: Degrees – Range: 0 to 360
snr	• SV signal-to-noise ratio – Units: dB-Hz

8.71.2 Field Documentation

8.71.2.1 float loc_satelliteInfo::azimuth

8.71.2.2 float loc_satelliteInfo::elevation

8.71.2.3 uint16_t loc_satelliteInfo::gnssSvId

8.71.2.4 uint8_t loc_satelliteInfo::healthStatus

8.71.2.5 float loc_satelliteInfo::snr

8.71.2.6 uint8_t loc_satelliteInfo::svInfoMask

8.71.2.7 uint8_t loc_satelliteInfo::svListLen

8.71.2.8 uint32_t loc_satelliteInfo::svStatus

8.71.2.9 uint32_t loc_satelliteInfo::system

8.71.2.10 uint32_t loc_satelliteInfo::validMask

8.72 loc_sensorDataUsage Struct Reference

Data Fields

- uint32_t usageMask
- uint32_t aidingIndicatorMask

8.72.1 Detailed Description

This structure contains Sensor Data Usage info.

Parameters

usageMask	• Specifies which sensors were used in calculating the position in the position report. – Value * 0x00000001 - Accelerometer used * 0x00000002 - Gyroscope used
aidingIndicator-Mask	• Specifies which results were aided by sensors. – Value * 0x00000001 - AIDED_HEADING * 0x00000002 - AIDED_SPEED * 0x00000004 - AIDED_POSITION * 0x00000008 - AIDED_VELOCITY

8.72.2 Field Documentation

8.72.2.1 uint32_t loc_sensorDataUsage::aidingIndicatorMask

8.72.2.2 uint32_t loc_sensorDataUsage::usageMask

8.73 loc_SV Struct Reference

Data Fields

- uint16_t id
- uint32_t system
- uint8_t mask

8.73.1 Detailed Description

This structure contains the Delete LOC SV Info

Parameters

id	• LOC SV ID of the satellite whose data is to be deleted • Range: – For GPS: 1 to 32 – For SBAS: 33 to 64 – For GLONASS: 65 to 96
system	• Indicates to which constellation this loc_SV belongs • Valid values: – eQMI_LOC_SV_SYSTEM_GPS (1) - GPS satellite – eQMI_LOC_SV_SYSTEM_GALILEO (2) - GALILEO satellite – eQMI_LOC_SV_SYSTEM_SBAS (3) - SBAS satellite – eQMI_LOC_SV_SYSTEM_COMPASS (4) - COMPASS satellite – eQMI_LOC_SV_SYSTEM_GLONASS (5) - GLONASS satellite – eQMI_LOC_SV_SYSTEM_BDS (6) - BDS satellite

mask	• Indicates if the ephemeris or almanac for a satellite is to be deleted • Valid values: – 0x01 - DELETE_EPHEMERIS – 0x02 - DELETE_ALMANAC

8.73.2 Field Documentation

8.73.2.1 uint16_t loc_SV::id

8.73.2.2 uint8_t loc_SV::mask

8.73.2.3 uint32_t loc_SV::system

8.74 loc_SVInfo Struct Reference

Data Fields

- uint8_t len
- loc_SV * pSV

8.74.1 Detailed Description

This structure contains the elements of Delete LOC SV Info

Parameters

len	• Number of sets of the following elements in struct loc_SV: – gnssSvId – system – deleteSvInfoMask
pSV	• Pointer to struct loc_SV. See loc_SV for more information

8.74.2 Field Documentation

8.74.2.1 uint8_t loc_SVInfo::len

8.74.2.2 loc_SV* loc_SVInfo::pSV

8.75 loc_svUsedforFix Struct Reference

Data Fields

- uint8_t gnssSvUsedList_len
- uint16_t gnssSvUsedList [255]

8.75.1 Detailed Description

This structure contains SVs Used to Calculate the Fix.

Parameters

gnssSvUsedList-_len	• Number of sets of gnssSvUsedList
pGnssSvUsed-List	• Entry in the list contains the SV ID of a satellite used for calculating this position report. • Following information is associated with each SV ID: – GPS - 1 to 32 – SBAS - 33 to 64 – GLONASS - 65 to 96 – QZSS - 193 to 197 – BDS - 201 to 237

8.75.2 Field Documentation

8.75.2.1 uint16_t loc_svUsedforFix::gnssSvUsedList[255]

8.75.2.2 uint8_t loc_svUsedforFix::gnssSvUsedList_len

8.76 loc_urlAddr Struct Reference

Data Fields

- char address [256]
- uint TlvPresent

8.76.1 Detailed Description

This structure contains LOC Get Server URL address field.

Parameters

address	• Uniform Resource Locator • Type - NULL terminated string • Maximum string length (including NULL terminator) 256
TlvPresent	• Tlv Present or not

8.76.2 Field Documentation

8.76.2.1 char loc_urlAddr::address[256]

8.76.2.2 uint loc_urlAddr::TlvPresent

8.77 lteSSInfo Struct Reference

Data Fields

- int8_t rssi
- int8_t rsrq
- int16_t rsrp
- int16_t snr

8.77.1 Detailed Description

This structure contains the parameters for LTE Signal Strength Information

Parameters

rssi	• RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal. – For CDMA and UMTS, this indicates forward link pilot Ec – For GSM, this indicates received signal strength
rsrq	• RSRQ value in dB (signed integer value) as measured by L1. • Range: -3 to -20 (-3 means -3 dB, -20 means -20 dB).
rsrp	• Current RSRP in dBm as measured by L1. • Range: -44 to -140 (-44 means -44 dBm, -140 means -140 dBm).
snr	• SNR level as a scaled integer in units of 0.1 dB. e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246,

8.77.2 Field Documentation

8.77.2.1 int16_t lteSSInfo::rsrp

8.77.2.2 int8_t lteSSInfo::rsrq

8.77.2.3 int8_t lteSSInfo::rssi

8.77.2.4 int16_t lteSSInfo::snr

8.78 messageModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMessageModeInfo MessageModeInfo

8.78.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
MessageMode-Info	• Message Mode • See sMSMessageModeInfo for more information

8.78.2 Field Documentation

8.78.2.1 sMSMessageModeInfo messageModeTlv::MessageModeInfo

8.78.2.2 uint8_t messageModeTlv::TlvPresent

8.79 nas_acqOrderPref Struct Reference

Data Fields

- uint8_t acqOrdeLen
- uint8_t * pAcqOrder

8.79.1 Detailed Description

Contain the Acquisition Order Preference.

Parameters

acqOrdeLen	• Number of sets of the following elements.
pAcqOrder	• Acquisition order preference to be set. Values: – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA - TD-SCDMA

8.79.2 Field Documentation

8.79.2.1 uint8_t nas_acqOrderPref::acqOrdeLen

8.79.2.2 uint8_t* nas_acqOrderPref::pAcqOrder

8.80 nas_ActPilotPNElement Struct Reference

Data Fields

- uint16_t ActSetPilotPN

- uint8_t ActSetPilotPNStrength

8.80.1 Detailed Description

Parameters

ActSetPilotPN	• Active Pilot PN
ActSetPilotPN-Strength	• Active Pilot PN strength

8.80.2 Field Documentation

8.80.2.1 uint16_t nas_ActPilotPNElement::ActSetPilotPN

8.80.2.2 uint8_t nas_ActPilotPNElement::ActSetPilotPNStrength

8.81 nas_AddCDMASysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx
- uint16_t regPrd

8.81.1 Detailed Description

Structure for storing the Additional CDMA System Information.

Parameters

geoSysIdx	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
regPrd	• Registration period after the CDMA system is acquired. • When the CDMA registration period is not valid, 0xFFFF is used.

8.81.2 Field Documentation

8.81.2.1 uint16_t nas_AddCDMASysInfo::geoSysIdx

8.81.2.2 uint16_t nas_AddCDMASysInfo::regPrd

8.82 nas_AddSysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx
- uint32_t cellBroadcastCap

8.82.1 Detailed Description

Structure for storing the Additional GSM and WCDMA System Information.

Parameters

geoSysIdx	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
cellBroadcast-Cap	• Cell broadcast capability of the serving system. • When the CDMA registration period is not valid, 0xFFFF is used. – NAS_CELL_BROADCAST_CAP_UNKNOWN - Cell broadcast support is unknown – NAS_CELL_BROADCAST_CAP_OFF - Cell broadcast is not supported – NAS_CELL_BROADCAST_CAP_ON - Cell broadcast is supported

8.82.2 Field Documentation

8.82.2.1 uint32_t nas_AddSysInfo::cellBroadcastCap

8.82.2.2 uint16_t nas_AddSysInfo::geoSysIdx

8.83 nas_CallBarringSysInfo Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.83.1 Detailed Description

Structure for storing the GSM and WCDMA Call Barring System Information.

Parameters

csBarStatus	• Call barring status for circuit-switched calls. – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

psBarStatus	• Call barring status for packet-switched calls. – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.83.2 Field Documentation

8.83.2.1 uint32_t nas_CallBarringSysInfo::csBarStatus

8.83.2.2 uint32_t nas_CallBarringSysInfo::psBarStatus

8.84 nas_callBarStatus Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.84.1 Detailed Description

This structure contains Call Barring Status.

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

csBarStatus	• Call Barring Status for circuit-switched calls. • Values: • NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only • NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only • NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type • NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types • NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown
psBarStatus	• Call Barring Status for packet-switched calls. • Values: – NAS_CELL_ACCESS_NORMAL_ONLY - Cell access is allowed for normal calls only – NAS_CELL_ACCESS_EMERGENCY_ONLY - Cell access is allowed for emergency calls only – NAS_CELL_ACCESS_NO_CALLS - Cell access is not allowed for any call type – NAS_CELL_ACCESS_ALL_CALLS - Cell access is allowed for all call types – NAS_CELL_ACCESS_UNKNOWN - Cell access type is unknown

8.84.2 Field Documentation

8.84.2.1 uint32_t nas_callBarStatus::csBarStatus

8.84.2.2 uint32_t nas_callBarStatus::psBarStatus

8.85 nas_CDMAChannel Struct Reference

Data Fields

- uint16_t priChA
- uint16_t priChB
- uint16_t secChA
- uint16_t secChB

8.85.1 Detailed Description

Parameters

priChA	• A Channel number for the primary carrier. – 0xFFFF - Not Available
priChB	• B Channel number for the primary carrier. – 0xFFFF - Not Available
secChA	• A Channel number for the secondary carrier. – 0xFFFF - Not Available
secChB	• B Channel number for the secondary carrier. – 0xFFFF - Not Available

8.85.2 Field Documentation

8.85.2.1 uint16_t nas_CDMAChannel::priChA

8.85.2.2 uint16_t nas_CDMAChannel::priChB

8.85.2.3 uint16_t nas_CDMAChannel::secChA

8.85.2.4 uint16_t nas_CDMAChannel::secChB

8.86 nas_CDMAECIOThresh Struct Reference

Data Fields

- uint8_t CDMAECIOThreshListLen
- int16_t * pCDMAECIOThreshList

8.86.1 Detailed Description

This structure contains CDMA ECIO threshold related parameters.

Parameters

CDMAECIO-ThreshListLen	• Length of the CDMA ECIO threshold list parameter to follow
pCDMAECIO-ThreshList	• Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values. Range for ECIO values: -31.5 to 0 (in dB)

8.86.2 Field Documentation

8.86.2.1 uint8_t nas_CDMAECIOThresh::CDMAECIOThreshListLen

8.86.2.2 int16_t* nas_CDMAECIOThresh::pCDMAECIOThreshList

8.87 nas_CDMAInfo Struct Reference

Data Fields

- uint16_t sid
- uint16_t nid
- uint16_t baseId
- uint16_t refpn
- uint32_t baseLat
- uint32_t baseLong

8.87.1 Detailed Description

This structure contains information about the CDMA Network.

Parameters

sid	• System ID. – 0xFFFF - Not Available
nid	• Network ID. – 0xFFFF - Not Available
baseId	• Base station ID. – 0xFFFF - Not Available
refpn	• Reference PN. – 0xFFFF - Not Available

baseLat	• Latitude of the current base station in units of 0.25 sec. – 0xFFFFFFFF - Not Available
baseLong	• Longitude of the current base station in units of 0.25 sec. – 0xFFFFFFFF - Not Available

8.87.2 Field Documentation

8.87.2.1 uint16_t nas_CDMAInfo::baseId

8.87.2.2 uint32_t nas_CDMAInfo::baseLat

8.87.2.3 uint32_t nas_CDMAInfo::baseLong

8.87.2.4 uint16_t nas_CDMAInfo::nid

8.87.2.5 uint16_t nas_CDMAInfo::refpn

8.87.2.6 uint16_t nas_CDMAInfo::sid

8.88 nas_CDMARSSIThresh Struct Reference

Data Fields

- uint8_t CDMARSSIThreshListLen
- int16_t * pCDMARSSIThreshList

8.88.1 Detailed Description

This structure contains CDMA RSSI threshold related parameters.

Parameters

CDMARSSI-ThreshListLen	• Length of the CDMARSSI threshold list parameter to follow
pCDMARSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • maximum of 32 values. • Range for RSSI values:-105 to -21 (in dBm).

8.88.2 Field Documentation

8.88.2.1 uint8_t nas_CDMARSSIThresh::CDMARSSIThreshListLen

8.88.2.2 int16_t* nas_CDMARSSIThresh::pCDMARSSIThreshList

8.89 nas_CDMASysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoCDMA
- uint8_t isSysPrlMatchValid
- uint8_t isSysPrlMatch
- uint8_t pRevInUseValid
- uint8_t pRevInUse
- uint8_t bsPRevValid
- uint8_t bsPRev
- uint8_t ccsSupportedValid
- uint8_t ccsSupported
- uint8_t cdmaSysIdValid
- uint16_t systemID
- uint16_t networkID
- uint8_t bsInfoValid
- uint16_t baseId
- uint32_t baseLat
- uint32_t baseLong
- uint8_t packetZoneValid
- uint16_t packetZone
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]

8.89.1 Detailed Description

Structure for storing the CDMA System Information.

Parameters

sysInfoCDMA	• See nas_sysInfoCommon for more information.
isSysPrlMatch-Valid	• Indicates whether the system PRL match is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
isSysPrlMatch	• Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, roam_status carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, roam_status is set to the value based on the standard specification.

pRevInUseValid	• Indicates whether the P_Rev in use is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
pRevInUse	• The lesser of the base station P_Rev and mobile P_Rev • Only applicable for CDMA. – 0xFF - Not Available
bsPRevValid	• Indicates whether the base station P_Rev is valid – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
bsPRev	• Base station P_Rev. • Only applicable for CDMA. – 0xFF - Not Available
ccsSupported-Valid	• Indicates whether the supported concurrent service is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
ccsSupported	• Whether concurrent service is supported. • Only applicable for CDMA. – 0x00 - Not supported – 0x01 - Supported – 0xFF - Not Available
cdmaSysIdValid	• Indicates whether the CDMA system ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
systemID	• System ID. – 0xFFFF - Not Available
networkID	• Network ID. – 0xFFFF - Not Available

bsInfoValid	• Indicates whether the base station information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
baseLat	• Base station latitude in units of 0.25 sec. • Expressed as a two's complement signed number with positive numbers signifying North latitudes. – 0xFFFFFFFF - Not Available
baseLong	• Base station longitude in units of 0.25 sec. • Expressed as a two's complement signed number with positive numbers signifying East latitudes. – 0xFFFFFFFF - Not Available
packetZoneValid	• Indicates whether the packet zone is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
packetZone	• Packet zone (8-bit). – 0xFFFF indicates no packet zone. • Only applicable for CDMA.
networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
MCC[NAS_PLMN_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters • MCC wildcard value is returned as {'3', 0xFF, 0xFF}.
MNC[NAS_PLMN_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • MNC wildcard value is returned as {'7', 0xFF, 0xFF}.

8.89.2 Field Documentation

8.89.2.1 uint16_t nas_CDMASysInfo::baseId

8.89.2.2 uint32_t nas_CDMASysInfo::baseLat

8.89.2.3 uint32_t nas_CDMASysInfo::baseLong

8.89.2.4 uint8_t nas_CDMASysInfo::bsInfoValid

8.89.2.5 uint8_t nas_CDMASysInfo::bsPRev

8.89.2.6 uint8_t nas_CDMASysInfo::bsPRevValid

8.89.2.7 uint8_t nas_CDMASysInfo::ccsSupported

8.89.2.8 uint8_t nas_CDMASysInfo::ccsSupportedValid

8.89.2.9 uint8_t nas_CDMASysInfo::cdmaSysIdValid

8.89.2.10 uint8_t nas_CDMASysInfo::isSysPrlMatch

8.89.2.11 uint8_t nas_CDMASysInfo::isSysPrlMatchValid

8.89.2.12 uint8_t nas_CDMASysInfo::MCC[3]

8.89.2.13 uint8_t nas_CDMASysInfo::MNC[3]

8.89.2.14 uint16_t nas_CDMASysInfo::networkID

8.89.2.15 uint8_t nas_CDMASysInfo::networkIdValid

8.89.2.16 uint16_t nas_CDMASysInfo::packetZone

8.89.2.17 uint8_t nas_CDMASysInfo::packetZoneValid

8.89.2.18 uint8_t nas_CDMASysInfo::pRevInUse

8.89.2.19 uint8_t nas_CDMASysInfo::pRevInUseValid

8.89.2.20 nas_sysInfoCommon nas_CDMASysInfo::sysInfoCDMA

8.89.2.21 uint16_t nas_CDMASysInfo::systemID

8.90 nas_CDMASysInfoExt Struct Reference

Data Fields

- uint16_t MCC
- uint8_t imsi_11_12

8.90.1 Detailed Description

This structure contains CDMA system information extension

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

MCC	• Mobile Country Code
imsi_11_12	• IMSI_11_12

8.90.2 Field Documentation

8.90.2.1 uint8_t nas_CDMASysInfoExt::imsi_11_12

8.90.2.2 uint16_t nas_CDMASysInfoExt::MCC

8.91 nas_cellParams Struct Reference

Data Fields

- uint16_t pci
- int16_t rsrq
- int16_t rsrp
- int16_t rssi
- int16_t srxlev

8.91.1 Detailed Description

This structure contains information about the Cell parameters.

Parameters

pci	• Physical cell ID. • Range: 0 to 503.
rsrq	• Current RSRQ in 1/10 dB as measured by L1. • Range: -20.0 dB to -3.0 dB.
rsrp	• Current RSRP in 1/10 dBm as measured by L1. • Range: -140.0 dBm to -44.0 dBm.
rssi	• Current RSSI in 1/10 dBm as measured by L1. • Range: -120.0 dBm to 0.
srxlev	• Cell selection Rx level (Srxlev) value. • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.91.2 Field Documentation

8.91.2.1 uint16_t nas_cellParams::pci

8.91.2.2 int16_t nas_cellParams::rsrp

8.91.2.3 int16_t nas_cellParams::rsrq

8.91.2.4 int16_t nas_cellParams::rssi

8.91.2.5 int16_t nas_cellParams::srxlev

8.92 nas_CommInfo Struct Reference

Data Fields

- int8_t temperature
- uint8_t modemMode
- uint8_t systemMode
- uint8_t imsRegState
- uint8_t psState

8.92.1 Detailed Description

Structure for storing the common information for the device.

Parameters

temperature	• Temperature. – 8-bit signed integer – 0xFF - Not Available.
modemMode	• Modem Operating Mode. – 0x00 - POWERING OFF – 0x01 - FACTORY TEST – 0x02 - OFFLINE – 0x03 - OFFLINE_AMPS – 0x04 - OFFLINE_CDMA – 0x05 - ONLINE – 0x06 - LOW POWER MODE – 0x07 - RESETTING – 0x08 - NETWORK TEST – 0x09 - OFFLINE REQUEST – 0x0A - PSEUDO ONLINE – 0x0B - RESETTING MODEM – 0xFF - Unknown

systemMode	• System Acquisition Mode. – 0x00 - No service – 0x01 - AMPS – 0x02 - CDMA – 0x03 - GSM – 0x04 - HDR – 0x05 - WCDMA – 0x06 - GPS – 0x08 - WLAN – 0x09 - LTE – 0xFF - Unknown
imsRegState	• IMS Registration State. – 0x00 - NO SRV – 0x01 - IN PROG – 0x02 - FAILED – 0x03 - LIMITED – 0x04 - FULL SRV – 0xFF - Unknown
psState	• PS Attach State. – 0x00 - Attached – 0x01 - Detached – 0xFF - Unknown

8.92.2 Field Documentation

8.92.2.1 uint8_t nas_CommInfo::imsRegState

8.92.2.2 uint8_t nas_CommInfo::modemMode

8.92.2.3 uint8_t nas_CommInfo::psState

8.92.2.4 uint8_t nas_CommInfo::systemMode

8.92.2.5 int8_t nas_CommInfo::temperature

8.93 nas_CSGID Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t mncPcsDigits
- uint32_t id
- uint8_t rat

8.93.1 Detailed Description

Contain the CSGID.

Parameters

mcc	• MCC value. Range 0 to 999
mnc	• MNC value. Range 0 to 999
mncPcsDigits	• TRUE - MNC is a three-digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 • FALSE - MNC is a two-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
id	• Closed subscriber group identifier.
rat	• Radio interface technology of the CSG network. Values: – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE – 0x09 - RADIO_IF_TDSCDMA - TDS

8.93.2 Field Documentation

8.93.2.1 uint32_t nas_CSGID::id

8.93.2.2 uint16_t nas_CSGID::mcc

8.93.2.3 uint16_t nas_CSGID::mnc

8.93.2.4 uint8_t nas_CSGID::mncPcsDigits

8.93.2.5 uint8_t nas_CSGID::rat

8.94 nas_currentPLMN Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t netDescrLength
- uint8_t netDescr [255]

8.94.1 Detailed Description

This structure contains the current PLMN parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

MCC	• mobile country code – A 16 bit representation of MCC – Range 0 to 999
MNC	• mobile network code – A 16 bit representation of MNC – Range 0 to 999
netDescrLength	• Length of Network description field • Defaults to zero
netDescr	• Network Description – optional string containing network name or description

8.94.2 Field Documentation

8.94.2.1 uint16_t nas_currentPLMN::MCC

8.94.2.2 uint16_t nas_currentPLMN::MNC

8.94.2.3 uint8_t nas_currentPLMN::netDescr[255]

8.94.2.4 uint8_t nas_currentPLMN::netDescrLength

8.95 nas_dataSrvCapabilities Struct Reference

Data Fields

- uint8_t dataCapabilitiesLen
- uint8_t dataCapabilities [32]

8.95.1 Detailed Description

This structure contains the data services capability

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

dataCapabilities-Len	• Length of data capabilities list • Defaults to zero

dataCapabilities	• List of data capabilities • Values: – 0x01 - GPRS – 0x02 - EDGE – 0x03 - HSDPA – 0x04 - HSUPA – 0x05 - WCDMA – 0x06 - CDMA – 0x07 - EV-DO Rev0 – 0x08 - EV-DO RevA – 0x09 - GSM – 0x0A - EV-DO Rev B – 0x0B - LTE – 0x0C - HSDPA+ – 0x0D - DC-HSDPA+

8.95.2 Field Documentation

8.95.2.1 uint8_t nas_dataSrvCapabilities::dataCapabilities[32]

8.95.2.2 uint8_t nas_dataSrvCapabilities::dataCapabilitiesLen

8.96 nas_DataStatusDetail Struct Reference

Data Fields

- uint32_t IPAddress
- uint8_t LastErrCode

8.96.1 Detailed Description

Parameters

IPAddress	• IP Address • 0xABCDEFGH - AB.CD.EF.GH • Example: – 0x12345678 - 18.52.86.120 0x12=18 0x34=52 0x56=86 0x78=120 • 0xFFFFFFFF - NA
LastErrCode	• MIP Error code – 0x00 - MIP_RRP_CODE_SUCCESS – 0x01 - MIP_RRP_CODE_SUCCESS_NO_SIM_BINDINGS – 0x40 - MIP_RRP_CODE_FAILURE_FA_REASON_UNSPECIFIED – 0x41 - MIP_RRP_CODE_FAILURE_FA_ADMIN_PROHIBITED – 0x42 - MIP_RRP_CODE_FAILURE_FA_INSUFFICIENT_RESOURCES – 0x43 - MIP_RRP_CODE_FAILURE_FA_MOBILE_NODE_FAILED_AUTH – 0x44 - MIP_RRP_CODE_FAILURE_FA_HA_FAILED_AUTH – 0x45 - MIP_RRP_CODE_FAILURE_FA_REQUESTED_LIFETIME_TOO_LONG – 0x46 - MIP_RRP_CODE_FAILURE_FA_MALFORMED_REQUEST – 0x47 - MIP_RRP_CODE_FAILURE_FA_MALFORMED_REPLY – 0x48 - MIP_RRP_CODE_FAILURE_FA_ENCAPSULATION_UNAVAILABLE – 0x49 - MIP_RRP_CODE_FAILURE_FA_VJHC_UNAVAILABLE – 0x4A - MIP_RRP_CODE_FAILURE_FA_CANT_REV_TUN – 0x4B - MIP_RRP_CODE_FAILURE_FA_MUST_REV_TUN – 0x4C - MIP_RRP_CODE_FAILURE_FA_BAD_TTL – 0x4D - MIP_RRP_CODE_FAILURE_INVALID_COA – 0x4F - MIP_RRP_CODE_FAILURE_FA_DELIVERY_STYLE_NOT_SUPPORTED – 0x59 - MIP_RRP_CODE_FAILURE_FA_VS_REASON – 0x61 - MIP_RRP_CODE_FAILURE_MISSING_NAI – 0x62 - MIP_RRP_CODE_FAILURE_MISSING_HA_ADDR – 0x63 - MIP_RRP_CODE_FAILURE_MISSING_HOMEADDR – 0x68 - MIP_RRP_CODE_FAILURE_UNKNOWN_CHALLENGE – 0x69 - MIP_RRP_CODE_FAILURE_MISSING_CHALLENGE – 0x6A - MIP_RRP_CODE_FAILURE_STALE_CHALLENGE – 0x6B - MIP_RRP_CODE_FAILURE_MISSING_MN_FA – 0x80 - MIP_RRP_CODE_FAILURE_HA_REASON_UNSPECIFIED – 0x81 - MIP_RRP_CODE_FAILURE_HA_ADMIN_PROHIBITED – 0x82 - MIP_RRP_CODE_FAILURE_HA_INSUFFICIENT_RESOURCES – 0x83 - MIP_RRP_CODE_FAILURE_HA_MOBILE_NODE_FAILED_AUTH – 0x84 - MIP_RRP_CODE_FAILURE_HA_FA_FAILED_AUTH – 0x85 - MIP_RRP_CODE_FAILURE_HA_REG_ID_MISMATCH – 0x86 - MIP_RRP_CODE_FAILURE_HA_MALFORMED_REQUEST – 0x88 - MIP_RRP_CODE_FAILURE_UNKNOWN_HA – 0x89 - MIP_RRP_CODE_FAILURE_HA_CANT_REV_TUN – 0x8A - MIP_RRP_CODE_FAILURE_HA_MUST_REV_TUN – 0x8B - MIP_RRP_CODE_FAILURE_HA_ENCAPSULATION_UNAVAILABLE – 0x8F - MIP_RRP_CODE_FAILURE_REDIRECTED_HA – 0x90 - MIP_RRP_CODE_FAILURE_HA_BAD_AAA_AUTH – 0xFF - NA

8.96.2 Field Documentation

8.96.2.1 uint32_t nas_DataStatusDetail::IPAddress

8.96.2.2 uint8_t nas_DataStatusDetail::LastErrCode

8.97 nas_detailSvcInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t srvCapability
- uint8_t hdrSrvStatus
- uint8_t hdrHybrid
- uint8_t isSysForbidden

8.97.1 Detailed Description

This structure contains Detailed Service information

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

srvStatus	• Service status • Values: – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep
srvCapability	• System's service capability • Values: – 0x00 - No Service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - MS found the right system but not yet registered/attached
hdrSrvStatus	• HDR service status • Values: – 0x00 - No service – 0x01 - Limited service – 0x02 - Service available – 0x03 - Limited regional service – 0x04 - MS in power save or deep sleep

hdrHybrid	• HDR hybrid information • Values: – 0x00 - System is not hybrid – 0x01 - System is hybrid
isSysForbidden	• Forbidden system information • Values: – 0x00 - System is not a forbidden system – 0x01 - System is a forbidden system

8.97.2 Field Documentation

8.97.2.1 uint8_t nas_detailSvcInfo::hdrHybrid

8.97.2.2 uint8_t nas_detailSvcInfo::hdrSrvStatus

8.97.2.3 uint8_t nas_detailSvcInfo::isSysForbidden

8.97.2.4 uint8_t nas_detailSvcInfo::srvCapability

8.97.2.5 uint8_t nas_detailSvcInfo::srvStatus

8.98 nas_DeviceConfigDetail Struct Reference

Data Fields

- uint8_t Technology
- uint8_t QLIC
- uint8_t Chipset
- uint8_t HWVersion

8.98.1 Detailed Description

Parameters

Technology	• Current technology in use – 0x00 - 1x RTT – 0x01 - EVDO Rev 0 – 0x02 - EVDO Rev A – 0xFF - NA
QLIC	• Quasi Linear Interference Cancellation – 0x00 - Not supported – 0x01 - Supported

Chipset	• Qualcomm platform – 0x4E - MDM6200 – 0x4F - MDM6600 – 0xFF - NA
HWVersion	• Hardware version – 0x00 - BSHWREV0 – 0x01 - BSHWREV1 – 0x02 - BSHWREV2 – 0x03 - BSHWREV3 – 0x04 - BSHWREVMAX – 0xFF - BSHWREVUNKNOWN

8.98.2 Field Documentation

8.98.2.1 uint8_t nas_DeviceConfigDetail::Chipset

8.98.2.2 uint8_t nas_DeviceConfigDetail::HWVersion

8.98.2.3 uint8_t nas_DeviceConfigDetail::QLIC

8.98.2.4 uint8_t nas_DeviceConfigDetail::Technology

8.99 nas_dirNum Struct Reference

Data Fields

- uint8_t dirNumLen
- uint8_t dirNum [255]

8.99.1 Detailed Description

This structure contains the parameters for Directory Number Information

Parameters

dirNumLen	• Number of sets of the following elements: – dir_num • If zero(0), then no information follows.
dirNum	• Directory number in ASCII characters.

8.99.2 Field Documentation

8.99.2.1 uint8_t nas_dirNum::dirNum[255]

8.99.2.2 uint8_t nas_dirNum::dirNumLen

8.100 nas_DRCParams Struct Reference

Data Fields

- uint8_t DRCValue
- uint8_t DRCCover

8.100.1 Detailed Description

Parameters

DRCValue	• Current Data Rate Channel
DRCCover	• Current Data Rate Channel cover

8.100.2 Field Documentation

8.100.2.1 uint8_t nas_DRCParams::DRCCover

8.100.2.2 uint8_t nas_DRCParams::DRCValue

8.101 nas_ecioListElement Struct Reference

Data Fields

- int16_t ecio
- uint8_t radioIf

8.101.1 Detailed Description

This structure contains the ECIO Information

Parameters

ecio	• ECIO value in dBm
radioIf	• Radio interface technology of the signal being measured – 0x00 – RADIO_IF_NO_SVC – None (no service) – 0x01 – RADIO_IF_CDMA_1X – cdma2000® 1X – 0x02 – RADIO_IF_CDMA_1XEVDO – cdma2000 HRPD (1xEV-DO) – 0x03 – RADIO_IF_AMPS – AMPS – 0x04 – RADIO_IF_GSM – GSM – 0x05 – RADIO_IF_UMTS – UMTS

8.101.2 Field Documentation

8.101.2.1 int16_t nas_ecioListElement::ecio

8.101.2.2 uint8_t nas_ecioListElement::radioIf

8.102 nas_ECIOThresh Struct Reference

Data Fields

- uint8_t ECIOThresListLen
- int16_t * pECIOThresList

8.102.1 Detailed Description

Parameters

ECIOThresList-Len	• Length of the ECIO threshold list parameter to follow
pECIOThresList	• Sequence of thresholds delimiting ECIO event reporting bands • Every time a new ECIO value crosses a threshold value, an event report indication message with the new ECIO value is sent to the requesting control point. For this field – Each ECIO threshold value is a signed 2 byte value – Each ECIO threshold value increments in negative 0.5 dB, e.g., an ECIO threshold value of 2 means -1dB – Maximum number of threshold values is 16 – At least one value must be specified – Threshold values specified above are used for all RATs

8.102.2 Field Documentation

8.102.2.1 uint8_t nas_ECIOThresh::ECIOThresListLen

8.102.2.2 int16_t* nas_ECIOThresh::pECIOThresList

8.103 nas_errorRateListElement Struct Reference

Data Fields

- uint16_t errorRate
- uint8_t radioIf

8.103.1 Detailed Description

This structure contains the Error Rate Information

Parameters

errorRate	• Error rate value corresponds to the RAT that is currently registered. – For CDMA, the error rate reported is Frame Error Rate: * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For HDR, the error rate reported is Packet Error Rate: * Valid error rate values between 1 and 10000 are returned to indicate percentage, e.g., a value of 300 means the error rate is 3% * A value of 0xFFFF indicates that the error rate is unknown or unavailable – For GSM, the error rate reported is Bit Error Rate: * Valid values are 0, 100, 200, 300, 400, 500, 600, and 700 The reported value divided by 100 gives the error rate as an RxQual value, e.g.,a value of 300 represents an RxQual value of 3. * A value of 25500 indicates No Data – For WCDMA, the error rate reported is Block Error Rate (BLER): * Valid values are 1 to 10000 * The reported value divided by 100 provides the error rate in percentages, e.g., a value of 300 represents a BLER of 3%. * A value of 0 indicates No Data
radioIf	• Radio interface technology of the signal being measured – 0x00 – RADIO_IF_NO_SVC – None (no service) – 0x01 – RADIO_IF_CDMA_1X – cdma2000® 1X – 0x02 – RADIO_IF_CDMA_1XEVDO – cdma2000 HRPD (1xEV-DO) – 0x03 – RADIO_IF_AMPS – AMPS – 0x04 – RADIO_IF_GSM – GSM – 0x05 – RADIO_IF_UMTS – UMTS

8.103.2 Field Documentation

8.103.2.1 uint16_t nas_errorRateListElement::errorRate

8.103.2.2 uint8_t nas_errorRateListElement::radioIf

8.104 nas_GERANInfo Struct Reference

Data Fields

- uint32_t cellID
- uint8_t plmn [3]
- uint16_t lac
- uint16_t arfcn
- uint8_t bsic
- uint32_t timingAdvance
- uint16_t rxLev
- uint8_t nmrInst
- nas_nmrCellInfo insNmrCellInfo [255]

8.104.1 Detailed Description

This structure contains information about the GERAN Network.

Parameters

cellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
plmn[NAS_PLMN_LENGTH]	• MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when nmrCellID is not present.
lac	• Location area code. • This field is ignored when nmrCellID is not present. – 0xFFFF - Not Available
arfcn	• Absolute RF channel number. – 0xFFFF - Not Available
bsic	• Base station identity code. – 0xFF - Not Available
timingAdvance	• Measured delay (in bit periods; 1 bit period = 48/13 microsecond) of access burst transmission on RACH or PRACH to the expected signal from an MS at zero distance under static channel conditions. – 0xFFFFFFFF - Not Available
rxLev	• Serving Cell Rx measurement. • Values range between 0 and 63. • Mapped to a measured signal level: – Rxlev 0 is a signal strength less than -110 dBm – Rxlev 1 is -110 dBm to -109 dBm – Rxlev 2 is -109 dBm to -108 dBm – ... – Rxlev 62 is -49 dBm to -48 dBm – Rxlev 63 is greater than -48 dBm – 0xFFFF - Not Available
nmrInst	• Provides the number of set of instances which follow. • If 0(zero), then no information follows it.
insNmrCellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_nmrCellInfo for more information.

8.104.2 Field Documentation

8.104.2.1 uint16_t nas_GERANInfo::arfcn

8.104.2.2 uint8_t nas_GERANInfo::bsic

8.104.2.3 uint32_t nas_GERANInfo::cellID

8.104.2.4 nas_nmrCellInfo nas_GERANInfo::insNmrCellInfo[255]

8.104.2.5 uint16_t nas_GERANInfo::lac

8.104.2.6 uint8_t nas_GERANInfo::nmrInst

8.104.2.7 uint8_t nas_GERANInfo::plmn[3]

8.104.2.8 uint16_t nas_GERANInfo::rxLev

8.104.2.9 uint32_t nas_GERANInfo::timingAdvance

8.105 nas_geranInstInfo Struct Reference

Data Fields

- uint16_t geranArfcn
- uint8_t geranBsicNcc
- uint8_t geranBsicBcc
- int16_t geranRssi

8.105.1 Detailed Description

This structure contains information about the GERAN Instances in UMTS Network.

Parameters

geranArfcn	• Absolute RF channel number.
geranBsicNcc	• Base station identity code network color code. • 0xFF indicates information is not present.
geranBsicBcc	• Base station identity code base station color code. • 0xFF indicates information is not present.
geranRssi	• Received signal strength indicator.

8.105.2 Field Documentation

8.105.2.1 uint16_t nas_geranInstInfo::geranArfcn

8.105.2.2 uint8_t nas_geranInstInfo::geranBsicBcc

8.105.2.3 uint8_t nas_geranInstInfo::geranBsicNcc

8.105.2.4 int16_t nas_geranInstInfo::geranRssi

8.106 nas_gsmCellInfo Struct Reference

Data Fields

- uint16_t arfcn
- uint8_t band1900
- uint8_t cellIdValid
- uint8_t bsicId
- int16_t rssi
- int16_t srxlev

8.106.1 Detailed Description

This structure contains information about the GSM Cell.

Parameters

arfcn	• GSM frequency being reported. • Range: 0 to 1023.
band1900	• Band indicator for the GSM ARFCN • This field is only valid if arfcn is in the overlapping region. • If TRUE and the cell is in the overlapping region, the ARFCN is on the 1900 band. • If FALSE, it is on the 1800 band.
cellIdValid	• Flag indicating whether the base station identity code ID is valid.
bsicId	• Base station identity code ID, including base station color code and network color code. • The lower 6 bits can be set to any value.
rssi	• Measured RSSI value in 1/10 dB. • Range: -200.0 dB to 0
srxlev	• Cell selection Rx level (Srxlev) value. • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.106.2 Field Documentation

8.106.2.1 uint16_t nas_gsmCellInfo::arfcn

8.106.2.2 uint8_t nas_gsmCellInfo::band1900

8.106.2.3 uint8_t nas_gsmCellInfo::bsicId

8.106.2.4 uint8_t nas_gsmCellInfo::cellIdValid

8.106.2.5 int16_t nas_gsmCellInfo::rssi

8.106.2.6 int16_t nas_gsmCellInfo::srxlev

8.107 nas_GSMRSSIThresh Struct Reference

Data Fields

- uint8_t GSMRSSIThreshListLen
- int16_t * pGSMRSSIThreshList

8.107.1 Detailed Description

This structure contains GSM RSSI threshold related parameters.

Parameters

GSMRSSI-ThreshListLen	• Length of the GSM RSSI threshold list parameter to follow
pGSMRSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSSI values: -111 to -48 (in dBm)

8.107.2 Field Documentation

8.107.2.1 uint8_t nas_GSMRSSIThresh::GSMRSSIThreshListLen

8.107.2.2 int16_t* nas_GSMRSSIThresh::pGSMRSSIThreshList

8.108 nas_GSMSrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t trueSrvStatus
- uint8_t isPrefDataPath

8.108.1 Detailed Description

Structure for storing the service status information for GSM, WCDMA and LTE networks.

Parameters

srvStatus	• Service status of the system. – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
trueSrvStatus	• True service status of the system. • Not applicable to CDMA/HDR. – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
isPrefDataPath	• Whether the RAT is the preferred data path. – 0x00 - Not preferred – 0x01 - Preferred – 0xFF - Not Available

8.108.2 Field Documentation

8.108.2.1 uint8_t nas_GSMSrvStatusInfo::isPrefDataPath

8.108.2.2 uint8_t nas_GSMSrvStatusInfo::srvStatus

8.108.2.3 uint8_t nas_GSMSrvStatusInfo::trueSrvStatus

8.109 nas_GSMSysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoGSM
- uint8_t lacValid
- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejCause
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]
- uint8_t egprsSuppValid
- uint8_t egprsSupp

- uint8_t dtmSuppValid
- uint8_t dtmSupp

8.109.1 Detailed Description

Structure for storing the GSM System Information.

Parameters

sysInfoGSM	• See nas_sysInfoCommon for more information.
lacValid	• Indicates whether the location area code is valid.. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
lac	• Location area code. • Only applies to 3GPP. – 0xFFFF - Not Available
cellIdValid	• Indicates whether the cell ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
cellId	• Cell ID. – 0xFFFFFFFF - Not Available
regRejectInfo-Valid	• Indicates whether the registration reject information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
rejectSrvDomain	• Type of service domain in which the registration is rejected. – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
rejCause	• Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. – 0xFF - Not Available

networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
MCC[NAS_PLMN_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters
MNC[NAS_PLMN_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
egprsSuppValid	• Indicates whether the EGPRS support is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
egprsSupp	• EGPRS support indication. • Only applicable for GSM. – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available
dtmSuppValid	• Indicates whether Dual Transfer mode support is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
dtmSupp	• Dual Transfer mode support indication. • Only applicable for GSM. – 0x00 - Not available – 0x01 - Available – 0xFF - Not Available

8.109.2 Field Documentation

8.109.2.1 uint32_t nas_GSMSysInfo::cellId

8.109.2.2 uint8_t nas_GSMSysInfo::cellIdValid

8.109.2.3 uint8_t nas_GSMSysInfo::dtmSupp

8.109.2.4 uint8_t nas_GSMSysInfo::dtmSuppValid

8.109.2.5 uint8_t nas_GSMSysInfo::egprsSupp

8.109.2.6 uint8_t nas_GSMSysInfo::egprsSuppValid

8.109.2.7 uint16_t nas_GSMSysInfo::lac

8.109.2.8 uint8_t nas_GSMSysInfo::lacValid

8.109.2.9 uint8_t nas_GSMSysInfo::MCC[3]

8.109.2.10 uint8_t nas_GSMSysInfo::MNC[3]

8.109.2.11 uint8_t nas_GSMSysInfo::networkIdValid

8.109.2.12 uint8_t nas_GSMSysInfo::regRejectInfoValid

8.109.2.13 uint8_t nas_GSMSysInfo::rejCause

8.109.2.14 uint8_t nas_GSMSysInfo::rejectSrvDomain

8.109.2.15 nas_sysInfoCommon nas_GSMSysInfo::sysInfoGSM

8.110 nas_HDRECIOThresh Struct Reference

Data Fields

- uint8_t HDRECIOThreshListLen
- int16_t * pHDRECIOThreshList

8.110.1 Detailed Description

This structure contains HDR ECIO threshold related parameters.

Parameters

HDRECIO-ThreshListLen	• Length of the HDR ECIO threshold list parameter to follow
pHDRECIO-ThreshList	• Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for ECIO values: -31.5 to 0 (in dB).

8.110.2 Field Documentation

8.110.2.1 uint8_t nas_HDRECIOThresh::HDRECIOThreshListLen

8.110.2.2 int16_t* nas_HDRECIOThresh::pHDRECIOThreshList

8.111 nas_HDRIOThresh Struct Reference

Data Fields

- uint8_t HDRIOThreshListLen
- int16_t * pHDRIOThreshList

8.111.1 Detailed Description

This structure contains HDR IO threshold related parameters.

Parameters

HDRIOThresh-ListLen	• Length of the HDR IO threshold list parameter to follow
pHDRIOThresh-List	• Array of IO thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for IO values: -128 to -13 (in dBm).

8.111.2 Field Documentation

8.111.2.1 uint8_t nas_HDRIOThresh::HDRIOThreshListLen

8.111.2.2 int16_t* nas_HDRIOThresh::pHDRIOThreshList

8.112 nas_HDRRSSIThresh Struct Reference

Data Fields

- uint8_t HDRRSSIThreshListLen
- int16_t * pHDRRSSIThreshList

8.112.1 Detailed Description

This structure contains HDR RSSI threshold related parameters.

Parameters

HDRRSSI-ThreshListLen	• Length of the HDR RSSI threshold list parameter to follow
pHDRRSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -118 to -13 (in dBm).

8.112.2 Field Documentation

8.112.2.1 uint8_t nas_HDRRSSIThresh::HDRRSSIThreshListLen

8.112.2.2 int16_t* nas_HDRRSSIThresh::pHDRRSSIThreshList

8.113 nas_HDRSINRThresh Struct Reference

Data Fields

- uint8_t HDRSINRThresListLen
- uint8_t * pHDRSINRThresList

8.113.1 Detailed Description

Parameters

HDRSINRThres-ListLen	• Length of the HDR SINR threshold list parameter to follow
pHDRSINR-ThresList	• Sequence of thresholds delimiting SINR event reporting bands • Every time a new SINR value crosses a threshold value, an event report indication message with the new SINR value is sent to the requesting control point. For this field – SINR is reported only for HDR – Each SINR threshold value is an unsigned 1 byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.113.2 Field Documentation

8.113.2.1 uint8_t nas_HDRSINRThresh::HDRSINRThresListLen

8.113.2.2 uint8_t* nas_HDRSINRThresh::pHDRSINRThresList

8.114 nas_HDRSINRThreshold Struct Reference

Data Fields

- uint8_t HDRSINRThreshListLen
- uint16_t * pHDRSINRThreshList

8.114.1 Detailed Description

This structure contains HDR SINR threshold related parameters.

Parameters

HDRSINR-ThreshListLen	• Length of the HDR ECIO threshold list parameter to follow
pHDRSINR-ThreshList	• Array of SINR level thresholds (in units of 1) • maximum of 32 values. • Valid levels are 0 to 8 – 0x00 - SINR_LEVEL_0 is -9 dB – 0x01 - SINR_LEVEL_1 is -6 dB – 0x02 - SINR_LEVEL_2 is -4.5 dB – 0x03 - SINR_LEVEL_3 is -3 dB – 0x04 - SINR_LEVEL_4 is -2 dB – 0x05 - SINR_LEVEL_5 is +1 dB – 0x06 - SINR_LEVEL_6 is +3 dB – 0x07 - SINR_LEVEL_7 is +6 dB – 0x08 - SINR_LEVEL_8 is +9 dB

8.114.2 Field Documentation

8.114.2.1 uint8_t nas_HDRSINRThreshold::HDRSINRThreshListLen

8.114.2.2 uint16_t* nas_HDRSINRThreshold::pHDRSINRThreshList

8.115 nas_HDRSysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoHDR
- uint8_t isSysPrlMatchValid
- uint8_t isSysPrlMatch
- uint8_t hdrPersonalityValid
- uint8_t hdrPersonality
- uint8_t hdrActiveProtValid
- uint8_t hdrActiveProt
- uint8_t is856SysIdValid
- uint8_t is856SysId [16]

8.115.1 Detailed Description

Structure for storing the HDR System Information.

Parameters

sysInfoHDR	• See nas_sysInfoCommon for more information.

isSysPrlMatch-Valid	• Indicates whether the system PRL match is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
isSysPrlMatch	• Indicates whether the system is in a PRL. • Only applies to CDMA/HDR. – 0x00 - System is not in a PRL – 0x01 - System is in a PRL – 0xFF - Not Available • If the system is not in a PRL, roam_status carries the value from the default roaming indicator in the PRL. • If the system is in a PRL, roam_status is set to the value based on the standard specification.
hdrPersonality-Valid	• Indicates whether the HDR personality is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
hdrPersonality	• HDR personality information. • Only applicable for HDR. – 0x00 - None – 0x02 - HRPD – 0x03 - eHRPD – 0xFF - Not Available
hdrActiveProt-Valid	• Indicates whether the HDR active protocol revision information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
hdrActiveProt	• HDR active protocol revision information . • Only applicable for HDR. – 0x00 - None – 0x02 - HDR Rel 0 – 0x03 - HDR Rel A – 0x04 - HDR Rel B – 0xFF - Not Available
is856SysIdValid	• Indicates whether the IS-856 system ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

is856SysId[SLQS_SYSTEM_ID_SIZE]	• IS-856 system ID. • Only applicable for HDR.

8.115.2 Field Documentation

8.115.2.1 uint8_t nas_HDRSysInfo::hdrActiveProt

8.115.2.2 uint8_t nas_HDRSysInfo::hdrActiveProtValid

8.115.2.3 uint8_t nas_HDRSysInfo::hdrPersonality

8.115.2.4 uint8_t nas_HDRSysInfo::hdrPersonalityValid

8.115.2.5 uint8_t nas_HDRSysInfo::is856SysId[16]

8.115.2.6 uint8_t nas_HDRSysInfo::is856SysIdValid

8.115.2.7 uint8_t nas_HDRSysInfo::isSysPrlMatch

8.115.2.8 uint8_t nas_HDRSysInfo::isSysPrlMatchValid

8.115.2.9 nas_sysInfoCommon nas_HDRSysInfo::sysInfoHDR

8.116 nas_homeSIDNID Struct Reference

Data Fields

- uint8_t numInstances
- nas_sidNid SidNid [255]

8.116.1 Detailed Description

Parameters

numInstances	• Number of sets of the following elements: – sid – nid • If zero(0), then no information follows.
SidNid	• See nas_sidNid for more information

8.116.2 Field Documentation

8.116.2.1 uint8_t nas_homeSIDNID::numInstances

8.116.2.2 nas_sidNid nas_homeSIDNID::SidNid[255]

8.117 nas_infoInterFreq Struct Reference

Data Fields

- uint16_t earfcn
- uint8_t threshXLow
- uint8_t threshXHigh
- uint8_t cell_resel_priority
- uint8_t cells_len
- nas_cellParams cellInterFreqParams [255]

8.117.1 Detailed Description

This structure contains information about the inter-frequency.

Parameters

earfcn	• E-UTRA absolute radio frequency channel number of the serving cell. • Range: 0 to 65535.
threshXLow	• Cell Srxlev low threshold. • Range: 0 to 31. • When the serving cell does not exceed thresh_serving_low, the value of an evaluated cell must be smaller than this value to be considered for re-selection.
threshXHigh	• Cell Srxlev high threshold. • Range: 0 to 31. • When the serving cell exceeds thresh_serving_low, the value of an evaluated cell must be greater than this value to be considered for re-selection.
cell_resel_priority	• Cell re-selection priority • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE.
cells_len	• Provides the number of set of cell params.
cellInterFreqParams[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_cellParams for more information.

8.117.2 Field Documentation

8.117.2.1 uint8_t nas_infoInterFreq::cell_resel_priority

8.117.2.2 nas_cellParams nas_infoInterFreq::cellInterFreqParams[255]

8.117.2.3 uint8_t nas_infoInterFreq::cells_len

8.117.2.4 uint16_t nas_infoInterFreq::earfcn

8.117.2.5 uint8_t nas_infoInterFreq::threshXHigh

8.117.2.6 uint8_t nas_infoInterFreq::threshXLow

8.118 nas_IOThresh Struct Reference

Data Fields

- uint8_t IOThresListLen
- int32_t * pIOThresList

8.118.1 Detailed Description

Parameters

IOThresListLen	• Length of the LTE SNR threshold list parameter to follow
pIOThresList	• Sequence of thresholds delimiting IO event reporting bands • Every time a new IO value crosses a threshold value, an event report indication message with the new IO value is sent to the requesting control point. For this field – IO is applicable only for HDR – Each IO threshold value is a signed 4 byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.118.2 Field Documentation

8.118.2.1 uint8_t nas_IOThresh::IOThresListLen

8.118.2.2 int32_t* nas_IOThresh::pIOThresList

8.119 nas_lteEARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn0
- uint32_t earfcn1

8.119.1 Detailed Description

Parameters

status	• 0 - Disable • 1 - Enable

earfcn0	• Primary DL EARFCN to which the UE is locked
earfcn1	• Secondary DL EARFCN to which the UE is locked • Note : Make earfcn1 value equal to earfcn0 if only one EARFCN is desired.

8.119.2 Field Documentation

8.119.2.1 uint32_t nas_lteEARFCN::earfcn0

8.119.2.2 uint32_t nas_lteEARFCN::earfcn1

8.119.2.3 uint8_t nas_lteEARFCN::status

8.120 nas_lteGsmCellInfo Struct Reference

Data Fields

- uint8_t cellReselPriority
- uint8_t threshGsmHigh
- uint8_t threshGsmLow
- uint8_t nccPermitted
- uint8_t cells_len
- nas_gsmCellInfo GsmCellInfo [255]

8.120.1 Detailed Description

This structure contains information about the LTE GSM Cell.

Parameters

cellReselPriority	• Priority of this frequency group. • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE.
threshGsmHigh	• Reselection threshold for high priority layers. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
threshGsmLow	• Reselection threshold for low priority layers. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.

nccPermitted	• Bitmask specifying whether a neighbor with a specific network color code is to be reported. • Range: 0 to 255. • Bit n set to 1 means a neighbor with NCC n must be included in the report. This flag is synonymous with a blacklist in other RATs. • This field is only valid when ue_in_idle is TRUE.
cells_len	• Provides the number of set of gsm cells.
GsmCellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_gsmCellInfo for more information.

8.120.2 Field Documentation

8.120.2.1 uint8_t nas_lteGsmCellInfo::cellReselPriority

8.120.2.2 uint8_t nas_lteGsmCellInfo::cells_len

8.120.2.3 nas_gsmCellInfo nas_lteGsmCellInfo::GsmCellInfo[255]

8.120.2.4 uint8_t nas_lteGsmCellInfo::nccPermitted

8.120.2.5 uint8_t nas_lteGsmCellInfo::threshGsmHigh

8.120.2.6 uint8_t nas_lteGsmCellInfo::threshGsmLow

8.121 nas_LTEInfo Struct Reference

Data Fields

- uint8_t band
- uint8_t bandwidth
- uint16_t RXChan
- uint16_t TXChan
- uint8_t emmState
- uint8_t emmSubState
- uint8_t emmConnState

8.121.1 Detailed Description

Structure for storing the LTE information for the device.

Parameters

band	• LTE Band – 1 ∼ 40 (Band in decimal) – 0xFF - Invalid

bandwidth	• BandWidth. – 0x00 - 1.4 MHz – 0x01 - 3 MHz – 0x02 - 5 MHz – 0x03 - 10 MHz – 0x04 - 15 MHz – 0x05 - 20 MHz – 0x06 - Invalid – 0xFF - Unknown
RXChan	• RX channel number in decimal – 0xFFFF - Not Available
TXChan	• TX channel number in decimal – 0xFFFF - Not Available
emmState	• EMM State. – 0x00 - Deregistered – 0x01 - Reg Initiated – 0x02 - Registered – 0x03 - TAU Initiated – 0x04 - SR Initiated – 0x05 - Dereg Initiated – 0x06 - Invalid – 0xFF - Unknown

emmSubState	• EMM Sub State. – 0xFF - NOT Applicable • When EMM_state is 0x00: – 0x00 - No IMSI – 0x01 - PLMN Search – 0x02 - Attach Needed – 0x03 - No Cell – 0x04 - Attaching – 0x05 - Normal Service – 0x06 - Limited Service – 0x07 - Waiting for PDN • When EMM_state is 0x01: – 0x00 - Waiting for NW – 0x01 - Waiting for ESM • When EMM_state is 0x02: – 0x00 - Normal Service – 0x01 - Update Needed – 0x02 - Attempt Update – 0x03 - No Cell – 0x04 - PLMN Search – 0x05 - Limited Service – 0x06 - MM Update – 0x07 - IMSI Detach – 0x08 - Waiting for ESM
emmConnState	• EMM Connected Mode State. – 0x00 - RRC Idle – 0x01 - Waiting RRC Cfm – 0x02 - RRC Connected – 0x03 - RRC Releasing – 0xFF - Unknown

8.121.2 Field Documentation

8.121.2.1 uint8_t nas_LTEInfo::band

8.121.2.2 uint8_t nas_LTEInfo::bandwidth

8.121.2.3 uint8_t nas_LTEInfo::emmConnState

8.121.2.4 uint8_t nas_LTEInfo::emmState

8.121.2.5 uint8_t nas_LTEInfo::emmSubState

8.121.2.6 uint16_t nas_LTEInfo::RXChan

8.121.2.7 uint16_t nas_LTEInfo::TXChan

8.122 nas_LTEInfoInterfreq Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t freqsLen
- nas_infoInterFreq InfoInterfreq [255]

8.122.1 Detailed Description

This structure contains information about the LTE Inter-Frequency Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available
freqsLen	• Provides the number of set of inter frequency information. • If 0(zero), then no information follows it.
InfoInterfreq[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_infoInterFreq for more information.

8.122.2 Field Documentation

8.122.2.1 uint8_t nas_LTEInfoInterfreq::freqsLen

8.122.2.2 nas_infoInterFreq nas_LTEInfoInterfreq::InfoInterfreq[255]

8.122.2.3 uint8_t nas_LTEInfoInterfreq::ueInIdle

8.123 nas_LTEInfoIntrafreq Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t plmn [3]
- uint16_t tac
- uint32_t globalCellId
- uint16_t earfcn
- uint16_t servingCellId
- uint8_t cellReselPriority
- uint8_t sNonIntraSearch
- uint8_t threshServingLow
- uint8_t sIntraSearch
- uint8_t cellsLen
- nas_cellParams CellParams [255]

8.123.1 Detailed Description

This structure contains information about the LTE Intra-Frequency Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available
plmn[NAS_PLMN_LENGTH]	• PLMN ID coded as octet 3, 4, and 5.
tac	• Tracking area code. – 0xFFFF - Not Available
globalCellId	• Global cell ID in the system information block. – 0xFFFFFFFF - Not Available
earfcn	• E-UTRA absolute radio frequency channel number of the serving cell. • Range: 0 to 65535. – 0xFFFF - Not Available
servingCellId	• LTE serving cell ID. • Range: 0 to 503. • This is the cell ID of the serving cell and can be found in the cell list. – 0xFFFF - Not Available
cellReselPriority	• Priority for serving frequency. • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available
sNonIntraSearch	• S non-intra search threshold to control non-intrafrequency searches. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available
threshServingLow	• Serving cell low threshold. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available

sIntraSearch	• S Intra search threshold. • Range: 0 to 31. • The current cell measurement must fall below this threshold to consider intrafrequency for reselection. • This field is only valid when ue_in_idle is TRUE. – 0xFF - Not Available
cellsLen	• Provides the number of set of cell params. • If 0(zero), then no information follows it.
CellParams[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_cellParams for more information.

8.123.2 Field Documentation

8.123.2.1 nas_cellParams nas_LTEInfoIntrafreq::CellParams[255]

8.123.2.2 uint8_t nas_LTEInfoIntrafreq::cellReselPriority

8.123.2.3 uint8_t nas_LTEInfoIntrafreq::cellsLen

8.123.2.4 uint16_t nas_LTEInfoIntrafreq::earfcn

8.123.2.5 uint32_t nas_LTEInfoIntrafreq::globalCellId

8.123.2.6 uint8_t nas_LTEInfoIntrafreq::plmn[3]

8.123.2.7 uint16_t nas_LTEInfoIntrafreq::servingCellId

8.123.2.8 uint8_t nas_LTEInfoIntrafreq::sIntraSearch

8.123.2.9 uint8_t nas_LTEInfoIntrafreq::sNonIntraSearch

8.123.2.10 uint16_t nas_LTEInfoIntrafreq::tac

8.123.2.11 uint8_t nas_LTEInfoIntrafreq::threshServingLow

8.123.2.12 uint8_t nas_LTEInfoIntrafreq::ueInIdle

8.124 nas_LTEInfoNeighboringGSM Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t freqsLen
- nas_lteGsmCellInfo LteGsmCellInfo [255]

8.124.1 Detailed Description

This structure contains information about the LTE Neighboring GSM Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available
freqsLen	• Provides the number of set of LTE GSM cell information. • If 0(zero), then no information follows it.
LteGsmCellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_lteGsmCellInfo for more information.

8.124.2 Field Documentation

8.124.2.1 uint8_t nas_LTEInfoNeighboringGSM::freqsLen

8.124.2.2 nas_lteGsmCellInfo nas_LTEInfoNeighboringGSM::LteGsmCellInfo[255]

8.124.2.3 uint8_t nas_LTEInfoNeighboringGSM::ueInIdle

8.125 nas_LTEInfoNeighboringWCDMA Struct Reference

Data Fields

- uint8_t ueInIdle
- uint8_t freqsLen
- nas_lteWcdmaCellInfo LTEWCDMACellInfo [255]

8.125.1 Detailed Description

This structure contains information about the LTE Neighboring WCDMA Network.

Parameters

ueInIdle	• TRUE if the UE is in Idle mode, otherwise FALSE. – 0xFF - Not Available
freqsLen	• Provides the number of set of LTE WCDMA cell information. • If 0(zero), then no information follows it.
LTEWCDMACellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_lteWcdmaCellInfo for more information.

8.125.2 Field Documentation

8.125.2.1 uint8_t nas_LTEInfoNeighboringWCDMA::freqsLen

8.125.2.2 nas_lteWcdmaCellInfo nas_LTEInfoNeighboringWCDMA::LTEWCDMACellInfo[255]

8.125.2.3 uint8_t nas_LTEInfoNeighboringWCDMA::ueInIdle

8.126 nas_ltePCI Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn
- uint32_t pci

8.126.1 Detailed Description

Parameters

status	• 0 - Disable • 1 - Enable
earfcn	• UARFCN to which UE is locked
pci	• PCI to which the UE is locked

8.126.2 Field Documentation

8.126.2.1 uint32_t nas_ltePCI::earfcn

8.126.2.2 uint32_t nas_ltePCI::pci

8.126.2.3 uint8_t nas_ltePCI::status

8.127 nas_lteRsrpinformation Struct Reference

Data Fields

- int16_t rsrplevel

8.127.1 Detailed Description

This structure contains the LTE RSRP Information

Parameters

rsrplevel	• LTE RSRP in dBm as a mesaured by L1. Range: -44 to -140(-44 means -44dBm, -140 means -140dBm).

8.127.2 Field Documentation

8.127.2.1 int16_t nas_lteRsrpinformation::rsrplevel

8.128 nas_LTERSRPThresh Struct Reference

Data Fields

- uint8_t LTERSRPThreshListLen
- int16_t * pLTERSRPThreshList

8.128.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

LTERSRP-ThreshListLen	• Length of the LTE RSRP threshold list parameter to follow
pLTERSRP-ThreshList	• Array of RSRP thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSRP values: -140 to -44 (in dBm).

8.128.2 Field Documentation

8.128.2.1 uint8_t nas_LTERSRPThresh::LTERSRPThreshListLen

8.128.2.2 int16_t* nas_LTERSRPThresh::pLTERSRPThreshList

8.129 nas_LTERSRQThresh Struct Reference

Data Fields

- uint8_t LTERSRQThreshListLen
- int16_t * pLTERSRQThreshList

8.129.1 Detailed Description

This structure contains LTE RSRQ threshold related parameters.

Parameters

LTERSRQ-ThreshListLen	• Length of the LTE RSRQ threshold list parameter to follow
pLTERSRQ-ThreshList	• Array of RSRQ thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSRQ values: -20 to -3 (in dBm)

8.129.2 Field Documentation

8.129.2.1 uint8_t nas_LTERSRQThresh::LTERSRQThreshListLen

8.129.2.2 int16_t* nas_LTERSRQThresh::pLTERSRQThreshList

8.130 nas_LTERSSIThresh Struct Reference

Data Fields

- uint8_t LTERSSIThreshListLen
- int16_t * pLTERSSIThreshList

8.130.1 Detailed Description

This structure contains LTE RSSI threshold related parameters.

Parameters

LTERSSI-ThreshListLen	• Length of the LTE RSSI threshold list parameter to follow
pLTERSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -120 to 0 (in dBm)

8.130.2 Field Documentation

8.130.2.1 uint8_t nas_LTERSSIThresh::LTERSSIThreshListLen

8.130.2.2 int16_t* nas_LTERSSIThresh::pLTERSSIThreshList

8.131 nas_LTESigRptCfg Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.131.1 Detailed Description

Parameters

rptRate	• Rate on how often the LTE signal must be checked for reporting • Values: – 0 - Report using the default configuration – 1 - Report every 1 sec – 2 - Report every 2 sec – 3 - Report every 3 sec – 4 - Report every 4 sec – 5 - Report every 5 sec
avgPeriod	• Averaging period to be used for the LTE signal • Values: – 0 - Average using the default configuration – 1 - Average over 1 sec – 2 - Average over 2 sec – 3 - Average over 3 sec – 4 - Average over 4 sec – 5 - Average over 5 sec – 6 - Average over 6 sec – 7 - Average over 7 sec – 8 - Average over 8 sec – 9 - Average over 9 sec – 10 - Average over 10 sec

8.131.2 Field Documentation

8.131.2.1 uint8_t nas_LTESigRptCfg::avgPeriod

8.131.2.2 uint8_t nas_LTESigRptCfg::rptRate

8.132 nas_LTESigRptConfig Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.132.1 Detailed Description

This structure contains LTE RSRP threshold related parameters.

Parameters

rptRate	• Rate on how often the LTE signal must be checked for reporting Values • 0 - Report using the default configuration • 1 - Report every 1 sec • 2 - Report every 2 sec • 3 - Report every 3 sec • 4 - Report every 4 sec • 5 - Report every 5 sec
avgPeriod	• Averaging period to be used for the LTE signal. • Values – 0 - Average using the default configuration – 1 - Average over 1 sec – 2 - Average over 2 sec – 3 - Average over 3 sec – 4 - Average over 4 sec – 5 - Average over 5 sec – 6 - Average over 6 sec – 7 - Average over 7 sec – 8 - Average over 8 sec – 9 - Average over 9 sec – 10 - Average over 10 sec

8.132.2 Field Documentation

8.132.2.1 uint8_t nas_LTESigRptConfig::avgPeriod

8.132.2.2 uint8_t nas_LTESigRptConfig::rptRate

8.133 nas_lteSnrinformation Struct Reference

Data Fields

- int16_t snrlevel

8.133.1 Detailed Description

This structure contains the LTE SNR Information

Parameters

snrlevel	• LTE SNR level as a scaled integer in units of 0.1dB e.g. -16dB has a value of -160 and 24.6dB has value of 246.

8.133.2 Field Documentation

8.133.2.1 int16_t nas_lteSnrinformation::snrlevel

8.134 nas_LTESNRThresh Struct Reference

Data Fields

- uint8_t LTESNRThresListLen
- int16_t * pLTESNRThresList

8.134.1 Detailed Description

Parameters

LTESNRThres-ListLen	• Length of the LTE SNR threshold list parameter to follow
pLTESNRThres-List	• Sequence of thresholds delimiting SNR event reporting bands • Every time a SNR value crosses a threshold value, an event report indication message with the new SNR value is sent to the requesting control point. For this field – For LTE, each SNR threshold value is a signed 2 Byte value – Maximum number of threshold values is 16 – At least one value must be specified – SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246

8.134.2 Field Documentation

8.134.2.1 uint8_t nas_LTESNRThresh::LTESNRThresListLen

8.134.2.2 int16_t* nas_LTESNRThresh::pLTESNRThresList

8.135 nas_LTESNRThreshold Struct Reference

Data Fields

- uint8_t LTESNRThreshListLen
- int16_t * pLTESNRThreshList

8.135.1 Detailed Description

This structure contains LTE SNR threshold related parameters.

Parameters

LTESNRThresh-ListLen	• Length of the LTE SNR threshold list parameter to follow
pLTESNR-ThreshList	• Array of SNR thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for SNR values: -20 to 30 (in dB).

8.135.2 Field Documentation

8.135.2.1 uint8_t nas_LTESNRThreshold::LTESNRThreshListLen

8.135.2.2 int16_t* nas_LTESNRThreshold::pLTESNRThreshList

8.136 nas_LTESysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoLTE
- uint8_t lacValid
- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejCause
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]
- uint8_t tacValid
- uint16_t tac

8.136.1 Detailed Description

Structure for storing the LTE System Information.

Parameters

sysInfoLTE	• See nas_sysInfoCommon for more information.
lacValid	• Indicates whether the location area code is valid.. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
lac	• Location area code. • Only applies to 3GPP. – 0xFFFF - Not Available

cellIdValid	• Indicates whether the cell ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
cellId	• Cell ID. – 0xFFFFFFFF - Not Available
regRejectInfo-Valid	• Indicates whether the registration reject information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
rejectSrvDomain	• Type of service domain in which the registration is rejected. – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
rejCause	• Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. – 0xFF - Not Available
networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
MCC[NAS_PLM-N_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters
MNC[NAS_PLM-N_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.

tacValid	• Indicates whether tracking area code is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
tac	• Tracking area code. • Only applicable for LTE. – 0xFFFF - Not Available

8.136.2 Field Documentation

8.136.2.1 uint32_t nas_LTESysInfo::cellId

8.136.2.2 uint8_t nas_LTESysInfo::cellIdValid

8.136.2.3 uint16_t nas_LTESysInfo::lac

8.136.2.4 uint8_t nas_LTESysInfo::lacValid

8.136.2.5 uint8_t nas_LTESysInfo::MCC[3]

8.136.2.6 uint8_t nas_LTESysInfo::MNC[3]

8.136.2.7 uint8_t nas_LTESysInfo::networkIdValid

8.136.2.8 uint8_t nas_LTESysInfo::regRejectInfoValid

8.136.2.9 uint8_t nas_LTESysInfo::rejCause

8.136.2.10 uint8_t nas_LTESysInfo::rejectSrvDomain

8.136.2.11 nas_sysInfoCommon nas_LTESysInfo::sysInfoLTE

8.136.2.12 uint16_t nas_LTESysInfo::tac

8.136.2.13 uint8_t nas_LTESysInfo::tacValid

8.137 nas_lteWcdmaCellInfo Struct Reference

Data Fields

- uint16_t uarfcn
- uint8_t cellReselPriority
- uint16_t threshXhigh
- uint16_t threshXlow
- uint8_t cellsLen
- nas_wcdmaCellInfo WCDMACellInfo [255]

8.137.1 Detailed Description

This structure contains information about the LTE WCDMA Cell.

Parameters

uarfcn	• WCDMA layer frequency. • Range: 0 to 16383.
cellReselPriority	• Cell re-selection priority. • Range: 0 to 7. • This field is only valid when ue_in_idle is TRUE.
threshXhigh	• Re-selection low threshold. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
threshXlow	• Re-selection high threshold. • Range: 0 to 31. • This field is only valid when ue_in_idle is TRUE.
cellsLen	• Provides the number of set of WCDMA cells.
WCDMACellInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_wcdmaCellInfo for more information.

8.137.2 Field Documentation

8.137.2.1 uint8_t nas_lteWcdmaCellInfo::cellReselPriority

8.137.2.2 uint8_t nas_lteWcdmaCellInfo::cellsLen

8.137.2.3 uint16_t nas_lteWcdmaCellInfo::threshXhigh

8.137.2.4 uint16_t nas_lteWcdmaCellInfo::threshXlow

8.137.2.5 uint16_t nas_lteWcdmaCellInfo::uarfcn

8.137.2.6 nas_wcdmaCellInfo nas_lteWcdmaCellInfo::WCDMACellInfo[255]

8.138 nas_minBasedIMSI Struct Reference

Data Fields

- uint8_t mccM [3]
- uint16_t imsiM1112
- uint8_t imsiMS1 [7]

- uint8_t imsiMS2 [3]

8.138.1 Detailed Description

Parameters

mccM	• ASCII character representation of MCC_M
imsiM1112	• ASCII character representation of IMSI_M_11_12 value – 0xFFFF - Not Available
imsiMS1	• ASCII character representation of IMSI_M_S1 value
imsiMS2	• ASCII character representation of IMSI_M_S2 value

8.138.2 Field Documentation

8.138.2.1 uint16_t nas_minBasedIMSI::imsiM1112

8.138.2.2 uint8_t nas_minBasedIMSI::imsiMS1[7]

8.138.2.3 uint8_t nas_minBasedIMSI::imsiMS2[3]

8.138.2.4 uint8_t nas_minBasedIMSI::mccM[3]

8.139 nas_MNRInfo Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint32_t rat

8.139.1 Detailed Description

Structure contains Manual Network Register Information parameters

Parameters

mcc	• A 16-bit integer representation of Mobile Country Code. Range - 0 to 999.
mnc	• A 16-bit integer representation of Mobile Network Code. Range - 0 to 999.

rat	• Radio access technology for which to register. – 0x04 - RADIO_IF_GSM – 0x05 - RADIO_IF_UMTS – 0x08 - RADIO_IF_LTE

8.139.2 Field Documentation

8.139.2.1 uint16_t nas_MNRInfo::mcc

8.139.2.2 uint16_t nas_MNRInfo::mnc

8.139.2.3 uint32_t nas_MNRInfo::rat

8.140 nas_namName Struct Reference

Data Fields

- uint8_t namNameLen
- uint8_t namName [12]

8.140.1 Detailed Description

Parameters

namNameLen	• Number of sets of the following elements: – nam_name • If zero(0), then no information follows.
namName	• Name information in ASCII. The maximum length of nam_name is 12.

8.140.2 Field Documentation

8.140.2.1 uint8_t nas_namName::namName[12]

8.140.2.2 uint8_t nas_namName::namNameLen

8.141 nas_netSelectionPref Struct Reference

Data Fields

- uint8_t netReg
- uint16_t mcc
- uint16_t mnc

8.141.1 Detailed Description

Contain the network selection preference.

Parameters

netReg	• specifies one of the following actions: – 0x00 - Automatic registration * Device registers according to its provisioning; mcc and mnc fields are ignored – 0x01 - Manual Registration * Device registers to specified network; mcc and mnc must contain valid values
mcc	• MCC value. Range 0 to 999
mnc	• MNC value. Range 0 to 999

8.141.2 Field Documentation

8.141.2.1 uint16_t nas_netSelectionPref::mcc

8.141.2.2 uint16_t nas_netSelectionPref::mnc

8.141.2.3 uint8_t nas_netSelectionPref::netReg

8.142 nas_NetworkStat1x Struct Reference

Data Fields

- uint8_t State
- uint16_t SO
- uint32_t RX_PWR
- uint16_t RX_EC_IO
- uint32_t TX_PWR
- uint8_t ActSetCnt
- nas_ActPilotPNElement * pActPilotPNElements
- uint8_t NeighborSetCnt
- uint16_t * pNeighborSetPilotPN

8.142.1 Detailed Description

Parameters

State	• CDMA current access state – 0x01 - Searching – 0x02 - Idle – 0x03 - Traffic – Others - NA

SO	• CDMA service option – 0xFFFF - Not in a call – 0x0001 - Basic Variable Rate Voice Service(8kbps) – 0x0002 - Mobile Station Loopback(8kbps) – 0x0003 - Enhanced Variable Rate Codec(EVRC) Voice Service(8kbps) – 0x0006 - Short message Services(Rate Set 1) – 0x0009 - Mobile Station Loopback(13kbps) – 0x000E - Short Message Service (Rate Set 2) – 0x0011 - High Rate Voice Service(13kbps) – 0x0020 - Test Data Service Option(TDSO) – 0x0021 - cdma2000 High Speed Packet Data Service, Internet or ISO Protocol Stack – 0x0044 - EVRC-B Voice Service(8 kbps) – 0x0046 - EVRC-WB Voice Service(8 kbps) – 0x0049 - Voice Echo mode supports smart blanking(EVRC-NW) – 0x004B - Enhanced loopback – 0x8000 - Proprietary Service Option (Qualcomm Inc.)
RX_PWR	• RX Pwr(dBm) – 0xABCD00EF - -ABCD.EF dBm – ABCD00EF should be transferred to decimal while displaying – Example: 0x12340056 - -4660.86dBm 0x1234 = 4660, 0x0056 = 86 – 0xFFFFFFFF - NA
RX_EC_IO	• RX EC/IO(dB) – 0xABCD - -AB.CD dB – ABCD should be transferred to decimal while displaying – Example: 0x1234 - -18.52dB 0x12 = 18, 0x34 = 52 – 0xFFFF - NA
TX_PWR	• TX PWR(dBm) – 0xFFFFFFFF - NA – Others - display actual value in decimal – Example: 0x1234 - -4660dBm 0x1234 = 4660
ActSetCnt	• Count of active pilot PN elements • As input specifies number of sets of parameter pActPilotElements for which memory has been assigned • As output specifies the actual number of sets of parameter pActPilotElements returned by device
pActPilotPN-Elements	• See nas_ActPilotPNElement for more information

NeighborSetCnt	• Count of neighbor pilot PN elements • As input specifies number of sets of parameter pNeighborSetPilotPN for which memory has been assigned • As output specifies the actual number of sets of parameter pNeighborSetPilotPN returned by device
pNeighborSet-PilotPN	• Neighbor pilot PN

8.142.2 Field Documentation

8.142.2.1 uint8_t nas_NetworkStat1x::ActSetCnt

8.142.2.2 uint8_t nas_NetworkStat1x::NeighborSetCnt

8.142.2.3 nas_ActPilotPNElement* nas_NetworkStat1x::pActPilotPNElements

8.142.2.4 uint16_t* nas_NetworkStat1x::pNeighborSetPilotPN

8.142.2.5 uint16_t nas_NetworkStat1x::RX_EC_IO

8.142.2.6 uint32_t nas_NetworkStat1x::RX_PWR

8.142.2.7 uint16_t nas_NetworkStat1x::SO

8.142.2.8 uint8_t nas_NetworkStat1x::State

8.142.2.9 uint32_t nas_NetworkStat1x::TX_PWR

8.143 nas_NetworkStatEVDO Struct Reference

Data Fields

- uint8_t State
- uint8_t MACIndex
- uint8_t SectorIDLen
- uint16_t * pSectorID
- uint16_t RX_PWR
- uint16_t PER
- uint16_t PilotEnergy
- uint8_t SNR

8.143.1 Detailed Description

Parameters

State	• EVDO network access state – 0x00 - Sleep – 0x01 - Searching – 0x02 - Idle – 0x03 - Active – 0xFF - NA
MACIndex	• HDR Mac index – 0xFF - NA – Others - Display the actual value in decimal – Example: 0x12 - 18 0x12 = 18
SectorIDLen	(IN/OUT) • Sector ID length • As input specifies length of parameter pSectorID for which memory has been assigned • As output specifies the actual length of parameter pSectorID returned by device
pSectorID	• Sector ID
RX_PWR	• TX PWR(dBm) – 0xABCD - -ABCD dBm – ABCD should be transferred to decimal while displaying – Example: 0x1234 - -4660dBm 0x1234 = 4660 – 0xFFFF - NA
PER	• HDR Packet Error Rate – 0xFFFF - Unknown – Others - display the actual value in decimal – Example: 0x1234 - -4660dBm 0x1234 = 4660
PilotEnergy	• Pilt Energy (dB) – 0xFFFF - NA – 0xABCD should be transferred to decimal while displaying – Example: 0x1234 - -4660dBm 0x1234 = 4660
SNR	• Signal to Noise ratio (dB)

8.143.2 Field Documentation

8.143.2.1 uint8_t nas_NetworkStatEVDO::MACIndex

8.143.2.2 uint16_t nas_NetworkStatEVDO::PER

8.143.2.3 uint16_t nas_NetworkStatEVDO::PilotEnergy

8.143.2.4 uint16_t* nas_NetworkStatEVDO::pSectorID

8.143.2.5 uint16_t nas_NetworkStatEVDO::RX_PWR

8.143.2.6 uint8_t nas_NetworkStatEVDO::SectorIDLen

8.143.2.7 uint8_t nas_NetworkStatEVDO::SNR

8.143.2.8 uint8_t nas_NetworkStatEVDO::State

8.144 nas_nmrCellInfo Struct Reference

Data Fields

- uint32_t nmrCellID
- uint8_t nmrPlmn [3]
- uint16_t nmrLac
- uint16_t nmrArfcn
- uint8_t nmrBsic
- uint16_t nmrRxLev

8.144.1 Detailed Description

This structure contains information about the Network Measurement Report (NMR) Cell Information.

Parameters

nmrCellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
nmrPlmn[NAS_PLMN_LENGTH]	• MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when nmrCellID is not present.
nmrLac	• Location area code. • This field is ignored when nmrCellID is not present. – 0xFFFF - Not Available
nmrArfcn	• Absolute RF channel number. – 0xFFFF - Not Available
nmrBsic	• Base station identity code. – 0xFF - Not Available

nmrRxLev	• Cell Rx measurement. • Values range between 0 and 63. • Mapped to a measured signal level: – Rxlev 0 is a signal strength less than -110 dBm – Rxlev 1 is -110 dBm to -109 dBm – Rxlev 2 is -109 dBm to -108 dBm – ... – Rxlev 62 is -49 dBm to -48 dBm – Rxlev 63 is greater than -48 dBm – 0xFFFF - Not Available

8.144.2 Field Documentation

8.144.2.1 uint16_t nas_nmrCellInfo::nmrArfcn

8.144.2.2 uint8_t nas_nmrCellInfo::nmrBsic

8.144.2.3 uint32_t nas_nmrCellInfo::nmrCellID

8.144.2.4 uint16_t nas_nmrCellInfo::nmrLac

8.144.2.5 uint8_t nas_nmrCellInfo::nmrPlmn[3]

8.144.2.6 uint16_t nas_nmrCellInfo::nmrRxLev

8.145 nas_operatorNameString Struct Reference

Data Fields

- uint8_t PLMNName [255]

8.145.1 Detailed Description

Parameters

PLMNName	• PLMN name must be coded in a default 7-bit alphabet with b8 set to 0.

8.145.2 Field Documentation

8.145.2.1 uint8_t nas_operatorNameString::PLMNName[255]

8.146 nas_OperatorPLMNData Struct Reference

Data Fields

- uint8_t mcc [3]
- uint8_t mnc [3]

- uint16_t lac1
- uint16_t lac2
- uint8_t PLMNRecID

8.146.1 Detailed Description

Parameters

mcc	• MCC in ASCII string (a value of D in any of the digits is to be used to indicate a "wild" value for that corresponding digit).
mnc	• MNC in ASCII string (a value of D in any of the digits is to be used to indicate a "wild" value for that corresponding digit; digit 3 in MNC is optional and when not present, will be set as ASCII F).
lac1	• Location area code 1.
lac2	• Location area code 1.
PLMNRecID	• PLMN network name record identifier.

8.146.2 Field Documentation

8.146.2.1 uint16_t nas_OperatorPLMNData::lac1

8.146.2.2 uint16_t nas_OperatorPLMNData::lac2

8.146.2.3 uint8_t nas_OperatorPLMNData::mcc[3]

8.146.2.4 uint8_t nas_OperatorPLMNData::mnc[3]

8.146.2.5 uint8_t nas_OperatorPLMNData::PLMNRecID

8.147 nas_operatorPLMNList Struct Reference

Data Fields

- uint16_t numInstance
- nas_OperatorPLMNData PLMNData [255]

8.147.1 Detailed Description

Parameters

numInstance	• Number of sets of the elements.

PLMNData	• Refer OperatorPLMNLData for details (Optional).

8.147.2 Field Documentation

8.147.2.1 uint16_t nas_operatorPLMNList::numInstance

8.147.2.2 nas_OperatorPLMNData nas_operatorPLMNList::PLMNData[255]

8.148 nas_PhyCaAggPcellInfo Struct Reference

Data Fields

- uint16_t pci
- uint16_t freq
- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- uint16_t iLTEbandValue
- uint8_t TlvPresent

8.148.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Pcell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
dl_bw_value	• Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.
TlvPresent	• Tlv Present.

8.148.2 Field Documentation

8.148.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggPcellInfo::dl_bw_value

8.148.2.2 uint16_t nas_PhyCaAggPcellInfo::freq

8.148.2.3 uint16_t nas_PhyCaAggPcellInfo::iLTEbandValue

8.148.2.4 uint16_t nas_PhyCaAggPcellInfo::pci

8.148.2.5 uint8_t nas_PhyCaAggPcellInfo::TlvPresent

8.149 nas_PhyCaAggScellDlBw Struct Reference

Data Fields

- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- uint8_t TlvPresent

8.149.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation Downlink Bandwidth of Scell.

Parameters

dl_bw_value	• Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.

8.149.2 Field Documentation

8.149.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellDlBw::dl_bw_value

8.149.2.2 uint8_t nas_PhyCaAggScellDlBw::TlvPresent

8.150 nas_PhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.150.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

scell_idx	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
TlvPresent	• Tlv Present.

8.150.2 Field Documentation

8.150.2.1 uint8_t nas_PhyCaAggScellIndex::scell_idx

8.150.2.2 uint8_t nas_PhyCaAggScellIndex::TlvPresent

8.151 nas_PhyCaAggScellIndType Struct Reference

Data Fields

- uint16_t pci
- uint16_t freq
- NAS_LTE_CPHY_SCELL_STATE_LITE scell_state
- uint8_t TlvPresent

8.151.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Indicator Type.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.
TlvPresent	• Tlv Present.

8.151.2 Field Documentation

8.151.2.1 uint16_t nas_PhyCaAggScellIndType::freq

8.151.2.2 uint16_t nas_PhyCaAggScellIndType::pci

8.151.2.3 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellIndType::scell_state

8.151.2.4 uint8_t nas_PhyCaAggScellIndType::TlvPresent

8.152 nas_PhyCaAggScellInfo Struct Reference

Data Fields

- uint16_t pci
- uint16_t freq

- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- uint16_t iLTEbandValue
- NAS_LTE_CPHY_SCELL_STATE_LITE scell_state
- uint8_t TlvPresent

8.152.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
dl_bw_value	• Downlink Bandwidth Values. • See NAS_LTE_CPHY_CA_BW_NRB_LITE for more information.

iLTEbandValue	• Band value. • Range for LTE Band class 120 to 160. – 120 - LTE E-UTRA Operating Band 1 – 121 - LTE E-UTRA Operating Band 2 – 122 - LTE E-UTRA Operating Band 3 – 123 - LTE E-UTRA Operating Band 4 – 124 - LTE E-UTRA Operating Band 5 – 125 - LTE E-UTRA Operating Band 6 – 126 - LTE E-UTRA Operating Band 7 – 127 - LTE E-UTRA Operating Band 8 – 128 - LTE E-UTRA Operating Band 9 – 129 - LTE E-UTRA Operating Band 10 – 130 - LTE E-UTRA Operating Band 11 – 131 - LTE E-UTRA Operating Band 12 – 132 - LTE E-UTRA Operating Band 13 – 133 - LTE E-UTRA Operating Band 14 – 134 - LTE E-UTRA Operating Band 17 – 135 - LTE E-UTRA Operating Band 33 – 136 - LTE E-UTRA Operating Band 34 – 137 - LTE E-UTRA Operating Band 35 – 138 - LTE E-UTRA Operating Band 36 – 139 - LTE E-UTRA Operating Band 37 – 140 - LTE E-UTRA Operating Band 38 – 141 - LTE E-UTRA Operating Band 39 – 142 - LTE E-UTRA Operating Band 40 – 143 - LTE E-UTRA Operating Band 18 – 144 - LTE E-UTRA Operating Band 19 – 145 - LTE E-UTRA Operating Band 20 – 146 - LTE E-UTRA Operating Band 21 – 147 - LTE E-UTRA Operating Band 24 – 148 - LTE E-UTRA Operating Band 25 – 149 - LTE E-UTRA Operating Band 41 – 150 - LTE E-UTRA Operating Band 42 – 151 - LTE E-UTRA Operating Band 43 – 152 - LTE E-UTRA Operating Band 23 – 153 - LTE E-UTRA Operating Band 26 – 154 - LTE E-UTRA Operating Band 32 – 155 - LTE E-UTRA Operating Band 125 – 156 - LTE E-UTRA Operating Band 126 – 157 - LTE E-UTRA Operating Band 127 – 158 - LTE E-UTRA Operating Band 28 – 159 - LTE E-UTRA Operating Band 29 – 160 - LTE E-UTRA Operating Band 30
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE_LITE for more information.

TlvPresent	• Tlv Present.

8.152.2 Field Documentation

8.152.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellInfo::dl_bw_value

8.152.2.2 uint16_t nas_PhyCaAggScellInfo::freq

8.152.2.3 uint16_t nas_PhyCaAggScellInfo::iLTEbandValue

8.152.2.4 uint16_t nas_PhyCaAggScellInfo::pci

8.152.2.5 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellInfo::scell_state

8.152.2.6 uint8_t nas_PhyCaAggScellInfo::TlvPresent

8.153 nas_PilotSetData Struct Reference

Data Fields

- uint8_t NumPilots
- nas_PilotSetParams * pPilotSetInfo

8.153.1 Detailed Description

Parameters

NumPilots	• Number of Pilot Sets • As input specifies number of sets of parameter pPilotSetInfo for which memory has been assigned • As output specifies the actual number of sets of parameter pPilotSetInfo returned by device
pPilotSetInfo	• Pilot Set Parameters • See nas_PilotSetParams for more information.

note A buffer under sized error is returned if the number of sets of pPilotSetInfo returned by the device is greater than the value in NumPilots input parameter.

8.153.2 Field Documentation

8.153.2.1 uint8_t nas_PilotSetData::NumPilots

8.153.2.2 nas_PilotSetParams* nas_PilotSetData::pPilotSetInfo

8.154 nas_PilotSetParams Struct Reference

Data Fields

- uint32_t PilotType
- uint16_t PilotPN
- uint16_t PilotStrength

8.154.1 Detailed Description

Parameters

PilotType	• 0x00 - NAS_HRPD_PILOT_CURR_ACT_PLT Current Active Pilot • 0x01 - NAS_HRPD_PILOT_NEIGHBOR_PLT Neighbor pilot information
PilotPN	• Pilot PN sequence offset index
PilotStrength	• Strength of the pilot (in dB)

8.154.2 Field Documentation

8.154.2.1 uint16_t nas_PilotSetParams::PilotPN

8.154.2.2 uint16_t nas_PilotSetParams::PilotStrength

8.154.2.3 uint32_t nas_PilotSetParams::PilotType

8.155 nas_PLMNNetworkName Struct Reference

Data Fields

- uint8_t numInstance
- nas_PLMNNetworkNameData PLMNNetName [255]

8.155.1 Detailed Description

Parameters

numInstance	• Number of sets of the elements.
PLMNNetName	• Refer nas_PLMNNetworkNameData for details (Optional).

8.155.2 Field Documentation

8.155.2.1 uint8_t nas_PLMNNetworkName::numInstance

8.155.2.2 nas_PLMNNetworkNameData nas_PLMNNetworkName::PLMNNetName[255]

8.156 nas_PLMNNetworkNameData Struct Reference

Data Fields

- uint8_t codingScheme
- uint8_t countryInitials
- uint8_t longNameSpareBits
- uint8_t shortNameSpareBits
- uint8_t longNameLen
- uint8_t longName [255]
- uint8_t shortNameLen
- uint8_t shortName [255]

8.156.1 Detailed Description

Parameters

codingScheme	• Coding scheme: – 0 - CODING_SCHEME_CELL_BROADCAST_GSM - Cell broadcast data coding scheme, GSM default alphabet, language unspecified;defined in 3GPP TS 23.038. – 1 - CODING_SCHEME_UCS2 - UCS2 (16 bit);defined in ISO/IEC 10646
countryInitials	• Country's initials: – 0 - COUNTRY_INITIALS_DO_NOT_ADD - MS should not add the letters for the country's initials to the text string. – 1 - COUNTRY_INITIALS_ADD - MS should add the letters for the country's initials and a separator, e.g., a space, to the text string.
longNameSpare-Bits	• Long Name Spare Bits: – 1 - SPARE_BITS_8 - Bit 8 is spare and set to 0 in octet n – 2 - SPARE_BITS_7_TO_8 - Bits 7 and 8 are spare and set to 0 in octet n. – 3 - SPARE_BITS_6_TO_8 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n. – 4 - SPARE_BITS_5_TO_8 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n. – 5 - SPARE_BITS_4_TO_8 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n. – 6 - SPARE_BITS_3_TO_8 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n. – 7 - SPARE_BITS_2_TO_8 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n. – 0 - SPARE_BITS_UNKNOWN - Carries no information about the number of spare bits in octet n.
shortName-SpareBits	• Short Name Spare Bits: – 1 - SPARE_BITS_8 - Bit 8 is spare and set to 0 in octet n. – 2 - SPARE_BITS_7_TO_8 - Bits 7 and 8 are spare and set to 0 in octet n. – 3 - SPARE_BITS_6_TO_8 - Bits 6 to 8 (inclusive) are spare and set to 0 in octet n. – 4 - SPARE_BITS_5_TO_8 - Bits 5 to 8 (inclusive) are spare and set to 0 in octet n. – 5 - SPARE_BITS_4_TO_8 - Bits 4 to 8 (inclusive) are spare and set to 0 in octet n. – 6 - SPARE_BITS_3_TO_8 - Bits 3 to 8 (inclusive) are spare and set to 0 in octet n. – 7 - SPARE_BITS_2_TO_8 - Bits 2 to 8 (inclusive) are spare and set to 0 in octet n. – 0 - SPARE_BITS_UNKNOWN - Carries no information about the number of spare bits in octet n.

longNameLen	• It provides the length of long name.
longName	• Long name string in coding_scheme.
shortNameLen	• It provides the length of short name.
shortName	• Short name string in coding_scheme.

8.156.2 Field Documentation

8.156.2.1 uint8_t nas_PLMNNetworkNameData::codingScheme

8.156.2.2 uint8_t nas_PLMNNetworkNameData::countryInitials

8.156.2.3 uint8_t nas_PLMNNetworkNameData::longName[255]

8.156.2.4 uint8_t nas_PLMNNetworkNameData::longNameLen

8.156.2.5 uint8_t nas_PLMNNetworkNameData::longNameSpareBits

8.156.2.6 uint8_t nas_PLMNNetworkNameData::shortName[255]

8.156.2.7 uint8_t nas_PLMNNetworkNameData::shortNameLen

8.156.2.8 uint8_t nas_PLMNNetworkNameData::shortNameSpareBits

8.157 nas_protocolSubtypeElement Struct Reference

Data Fields

- uint16_t PhysicalLayer
- uint16_t ControlMac
- uint16_t AccessMac
- uint16_t ForwardMac
- uint16_t ReverseMac
- uint16_t KeyExchange
- uint16_t AuthProt
- uint16_t EncryptProt
- uint16_t SecProt
- uint16_t IdleState
- uint16_t MultDisc
- uint16_t VirtStream

8.157.1 Detailed Description

Parameters

PhysicalLayer	• Specifies Physical Layer Protocol subtype
ControlMac	• Specifies Control Channel MAC Protocol subtype
AccessMac	• Specifies Access Channel MAC Protocol subtype
ForwardMac	• Specifies Forward Traffic Channel MAC Protocol subtype
ReverseMac	• Specifies Reverse Traffic Channel MAC Protocol subtype
KeyExchange	• Specifies Key exchange Protocol subtype
AuthProt	• Specifies Authentication Protocol subtype
EncryptProt	• Specifies Encryption Protocol subtype
SecProt	• Specifies Security Protocol subtype
IdleState	• Specifies Idle state Protocol subtype
MultDisc	• Specifies Generic multimode capability discovery Protocol subtype
VirtStream	• Specifies Generic Virtual Stream Protocol subtype

8.157.2 Field Documentation

8.157.2.1 uint16_t nas_protocolSubtypeElement::AccessMac

8.157.2.2 uint16_t nas_protocolSubtypeElement::AuthProt

8.157.2.3 uint16_t nas_protocolSubtypeElement::ControlMac

8.157.2.4 uint16_t nas_protocolSubtypeElement::EncryptProt

8.157.2.5 uint16_t nas_protocolSubtypeElement::ForwardMac

8.157.2.6 uint16_t nas_protocolSubtypeElement::IdleState

8.157.2.7 uint16_t nas_protocolSubtypeElement::KeyExchange

8.157.2.8 uint16_t nas_protocolSubtypeElement::MultDisc

8.157.2.9 uint16_t nas_protocolSubtypeElement::PhysicalLayer

8.157.2.10 uint16_t nas_protocolSubtypeElement::ReverseMac

8.157.2.11 uint16_t nas_protocolSubtypeElement::SecProt

8.157.2.12 uint16_t nas_protocolSubtypeElement::VirtStream

8.158 nas_qaQmi3Gpp2TimeZone Struct Reference

Data Fields

- uint8_t leapSeconds
- uint8_t localTimeOffset
- uint8_t daylightSavings

8.158.1 Detailed Description

This structure contains the 3GPP2TimeZone parameters

Parameters

leapSeconds	• leap seconds - Number of leap seconds since the start of CDMA system time.
localTimeOffset	• Local Time Offset - Offset of system time in units of 30 minutes; the value in this field conveys as 8 bit 2's compliment number.
daylightSavings	• Day Light Savings Indicator – 0x00 - OFF (daylight savings not in effect) – 0x01 - ON (daylight savings in effect)

8.158.2 Field Documentation

8.158.2.1 uint8_t nas_qaQmi3Gpp2TimeZone::daylightSavings

8.158.2.2 uint8_t nas_qaQmi3Gpp2TimeZone::leapSeconds

8.158.2.3 uint8_t nas_qaQmi3Gpp2TimeZone::localTimeOffset

8.159 nas_QmiNas3GppNetworkInfo Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint32_t InUse
- uint32_t Roaming

- uint32_t Forbidden
- uint32_t Preferred
- char Desription [255]

8.159.1 Detailed Description

Contain the 3GPP network information.

Parameters

MCC	• Mobile Country Code
MNC	• Mobile Network Code
InUse	• Is the Network the current serving Network – 0 - Unknown – 1 - Current serving network – 2 - Not current serving network, available
Roaming	• Home/Roam Status of the Network – 0 - Unknown – 1 - Home – 2 - Roam
Forbidden	• Is the Network in the forbidden network list – 0 - Unknown – 1 - Forbidden – 2 - Not Forbidden
Preferred	• Is the Network in the Preferred network list – 0 - Unknown – 1 - Preferred – 2 - Not Preferred
Description	• Network Name/Description • This is a NULL terminated string.

8.159.2 Field Documentation

8.159.2.1 char nas_QmiNas3GppNetworkInfo::Desription[255]

8.159.2.2 uint32_t nas_QmiNas3GppNetworkInfo::Forbidden

8.159.2.3 uint32_t nas_QmiNas3GppNetworkInfo::InUse

8.159.2.4 uint16_t nas_QmiNas3GppNetworkInfo::MCC

8.159.2.5 uint16_t nas_QmiNas3GppNetworkInfo::MNC

8.159.2.6 uint32_t nas_QmiNas3GppNetworkInfo::Preferred

8.159.2.7 uint32_t nas_QmiNas3GppNetworkInfo::Roaming

8.160 nas_QmiNas3GppNetworkRAT Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t RAT

8.160.1 Detailed Description

Contain the 3GPP radio access technology information.

Parameters

MCC	• Mobile Country Code
MNC	• Mobile Network Code
RAT	• Radio Access Technology – 0x04 - GERAN – 0x05 - UMTS – 0x08 - LTE – 0x09 - TD-SCDMA

8.160.2 Field Documentation

8.160.2.1 uint16_t nas_QmiNas3GppNetworkRAT::MCC

8.160.2.2 uint16_t nas_QmiNas3GppNetworkRAT::MNC

8.160.2.3 uint8_t nas_QmiNas3GppNetworkRAT::RAT

8.161 nas_QmisNasPcsDigit Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t includes_pcs_digit

8.161.1 Detailed Description

Contain the PCS Digit information

Parameters

MCC	• Mobile Country Code
MNC	• Mobile Network Code
includes_pcs_digit	• this field is use to interpret the length of corresponding MNC reported • 0x01 - MNC is a three-digit value • 0x00 - MNC is a two-digit value

8.161.2 Field Documentation

8.161.2.1 uint8_t nas_QmisNasPcsDigit::includes_pcs_digit

8.161.2.2 uint16_t nas_QmisNasPcsDigit::MCC

8.161.2.3 uint16_t nas_QmisNasPcsDigit::MNC

8.162 nas_RejectReasonTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t serviceDomain
- uint32_t rejectCause

8.162.1 Detailed Description

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
serviceDomain	service domain
rejectCause	cause of reject

8.162.2 Field Documentation

8.162.2.1 uint32_t nas_RejectReasonTlv::rejectCause

8.162.2.2 uint32_t nas_RejectReasonTlv::serviceDomain

8.162.2.3 uint8_t nas_RejectReasonTlv::TlvPresent

8.163 nas_RFInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t radioInterfaceSize
- uint32_t radioInterface [255]
- uint32_t activeBandClass [255]
- uint32_t activeChannel [255]

8.163.1 Detailed Description

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
radioInterface	radio interface technology of the signal being measured
activeBandClass	active band class
activeChannel	active channel

8.163.2 Field Documentation

8.163.2.1 uint32_t nas_RFInfoTlv::activeBandClass[255]

8.163.2.2 uint32_t nas_RFInfoTlv::activeChannel[255]

8.163.2.3 uint32_t nas_RFInfoTlv::radioInterface[255]

8.163.2.4 uint8_t nas_RFInfoTlv::radioInterfaceSize

8.163.2.5 uint8_t nas_RFInfoTlv::TlvPresent

8.164 nas_roamIndList Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t radioInterface [32]
- uint8_t roamIndicator [32]

8.164.1 Detailed Description

This structure contains the Roaming Indicator List

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

numInstances	• number of sets of radio interface currently in use and roaming indicator – defaults to zero

radioInterface	• Radio Interface currently in use • Values: – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE
roamIndicator	• Roaming Indicator • Values: – 0x00 - Roaming – 0x01 - Home

8.164.2 Field Documentation

8.164.2.1 uint8_t nas_roamIndList::numInstances

8.164.2.2 uint8_t nas_roamIndList::radioInterface[32]

8.164.2.3 uint8_t nas_roamIndList::roamIndicator[32]

8.165 nas_RSRPThresh Struct Reference

Data Fields

- uint8_t RSRPThresListLen
- int16_t * pRSRPThresList

8.165.1 Detailed Description

Parameters

RSRPThresList-Len	• Length of the LTE RSRP threshold list parameter to follow
pRSRPThresList	• Sequence of thresholds delimiting current RSRP event reporting bands • Every time a new RSRP value crosses a specified threshold value, an event report indication message with the new RSRQ value is sent to the requesting control point. For this field – RSRP values are applicable only for LTE – RSRP values are measured in dBm, with a range of -44 dBm to -140 dBm – Each RSRP threshold value is a signed byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.165.2 Field Documentation

8.165.2.1 int16_t* nas_RSRPThresh::pRSRPThresList

8.165.2.2 uint8_t nas_RSRPThresh::RSRPThresListLen

8.166 nas_rsrqInformation Struct Reference

Data Fields

- int8_t rsrq
- uint8_t radioIf

8.166.1 Detailed Description

This structure contains the RSRQ Information

Parameters

rsrq	• RSRQ value in dB (signed integer value); valid range is -3 to -20 (-3 means -3 dB, -20 means -20 dB)
radioIf	• Radio interface technology of the signal being measured – 0x08 – LTE

8.166.2 Field Documentation

8.166.2.1 uint8_t nas_rsrqInformation::radioIf

8.166.2.2 int8_t nas_rsrqInformation::rsrq

8.167 nas_RSRQThresh Struct Reference

Data Fields

- uint8_t RSRQThresListLen
- int8_t * pRSRQThresList

8.167.1 Detailed Description

Parameters

RSRQThresList-Len	• Length of the LTE RSRQ threshold list parameter to follow
pRSRQThresList	• Sequence of thresholds delimiting current RSRQ event reporting bands • Every time a new RSRQ value crosses a threshold value, an event report indication message with the new RSRQ value is sent to the requesting control point. For this field – RSRQ values are applicable only for LTE – RSRQ values are measured in dBm, with a range of -20 dBm to -3 dBm – Each RSRQ threshold value is a signed byte value – Maximum number of threshold values is 16 – At least one value must be specified

8.167.2 Field Documentation

8.167.2.1 int8_t* nas_RSRQThresh::pRSRQThresList

8.167.2.2 uint8_t nas_RSRQThresh::RSRQThresListLen

8.168 nas_RSSIThresh Struct Reference

Data Fields

- uint8_t RSSIThresListLen
- int8_t * pRSSIThresList

8.168.1 Detailed Description

Parameters

RSSIThresList-Len	• Length of the RSSI threshold list parameter to follow
pRSSIThresList	• RSSI in dBm(signed bytes) • A value of -125 dBm or lower is used to indicate No Signal • RSSI values have the following ranges (in dBm) – CDMA is -105 to -21 – HDR is -118 to -13 – GSM is -111 to -48 – WCDMA is -121 to 0 – LTE is -120 to 0 • Threshold values specified above are used for all RATs • The maximum number of threshold values is 16, each a signed byte value.

8.168.2 Field Documentation

8.168.2.1 int8_t* nas_RSSIThresh::pRSSIThresList

8.168.2.2 uint8_t nas_RSSIThresh::RSSIThresListLen

8.169 nas_rxInfo Struct Reference

Data Fields

- uint8_t isRadioTuned
- int32_t rxPower
- int32_t ecio
- int32_t rscp
- int32_t rsrp
- uint32_t phase

8.169.1 Detailed Description

Parameters

isRadioTuned	• Whether Rx is tuned to a channel: – 0x00 - Not tuned – 0x01 - Tuned – 0xFF - Not Available • If the radio is tuned, instantaneous values are set for the signal information fields below. • If the radio is not tuned, or is delayed or invalid, the values are set depending on each technology.
rx_pwr	• Rx power value in 1/10 dbm resolution.
ecio	• ECIO in 1/10 dbm; valid for CDMA, HDR, GSM, WCDMA, and LTE.
rscp	• Received signal code power in 1/10 dbm. • Valid for WCDMA.
rsrp	• Current reference signal received power in 1/10 dbm valid for LTE.
phase	• Phase in 1/100 degrees; valid for LTE. • When the phase is unknown, 0xFFFFFFFF is used.

8.169.2 Field Documentation

8.169.2.1 int32_t nas_rxInfo::ecio

8.169.2.2 uint8_t nas_rxInfo::isRadioTuned

8.169.2.3 uint32_t nas_rxInfo::phase

8.169.2.4 int32_t nas_rxInfo::rscp

8.169.2.5 int32_t nas_rxInfo::rsrp

8.169.2.6 int32_t nas_rxInfo::rxPower

8.170 nas_RxSigInfo Struct Reference

Data Fields

- uint8_t rxChainIndex
- uint8_t isRadioTuned
- int32_t rxPower
- int32_t rsrp

8.170.1 Detailed Description

This structure contains the parameters for Rx Signal Info.

Parameters

rxChainIndex	• Rx antenna path • Valid Values – 0 - Primary Rx – 1 - Diversity Rx
isRadioTuned	• Rx path is tuned to a channel or Not • Values – 0x00 - Not tuned – 0x01 - Tuned

Note

If the radio is tuned, the instantaneous values are set for the fields below. If the radio is not tuned, the values set below may be invalid.

Parameters

rxPower	• Rx power value in 1/10 dBm resolution
rsrp	• Current reference signal received power in 1/10 dBm resolution

8.170.2 Field Documentation

8.170.2.1 uint8_t nas_RxSigInfo::isRadioTuned

8.170.2.2 int32_t nas_RxSigInfo::rsrp

8.170.2.3 uint8_t nas_RxSigInfo::rxChainIndex

8.170.2.4 int32_t nas_RxSigInfo::rxPower

8.171 nas_rxSignalStrengthListElement Struct Reference

Data Fields

- int16_t rxSignalStrength
- uint8_t radioIf

8.171.1 Detailed Description

This structure contains the Received Signal Strength Information

Parameters

rxSignalStrength	• Received signal strength in dBm – For CDMA and UMTS, this indicates forward link pilotEc. – For GSM, the received signal strength. – For LTE, this indicates the total received wideband power observed by UE.
radioIf	• Radio interface technology of the signal being radio_if measured – 0x00 – RADIO_IF_NO_SVC – None (no service) – 0x01 – RADIO_IF_CDMA_1X – cdma2000® 1X – 0x02 – RADIO_IF_CDMA_1XEVDO – cdma2000 HRPD (1xEV-DO) – 0x03 – RADIO_IF_AMPS – AMPS – 0x04 – RADIO_IF_GSM – GSM – 0x05 – RADIO_IF_UMTS – UMTS – 0x08 – RADIO_IF_LTE – LTE

Note

First elemnet of the RSSI list always contains the current Signal strength and Radio Interface.

8.171.2 Field Documentation

8.171.2.1 uint8_t nas_rxSignalStrengthListElement::radioIf

8.171.2.2 int16_t nas_rxSignalStrengthListElement::rxSignalStrength

8.172 nas_SccRxInfo Struct Reference

Data Fields

- int32_t rsrq
- int16_t snr
- uint8_t numInstances
- nas_RxSigInfo sigInfo [255]
- uint8_t TlvPresent

8.172.1 Detailed Description

This structure contains information about the SccRxInfo parameters.

Parameters

rsrq	• Current reference signal • Receive quality in 1/10 dB resolution
snr	• Reference signal signal-to-noise ratio in dB. • Range -10 to 30
numInstances	• Number of sets of the following – rxChainIndex – isRadioTuned – rxPower – rsrp
sigInfo	• See nas_RxSigInfo for more information
TlvPresent	• Tlv Present.

8.172.2 Field Documentation

8.172.2.1 uint8_t nas_SccRxInfo::numInstances

8.172.2.2 int32_t nas_SccRxInfo::rsrq

8.172.2.3 nas_RxSigInfo nas_SccRxInfo::sigInfo[255]

8.172.2.4 int16_t nas_SccRxInfo::snr

8.172.2.5 uint8_t nas_SccRxInfo::TlvPresent

8.173 nas_serviceProviderName Struct Reference

Data Fields

- uint8_t displayCondition
- uint8_t spnLength
- uint8_t spn [255]

8.173.1 Detailed Description

Parameters

displayCondition	• Display condition.
spnLength	• It provides length of spn.
spn	• Service provider name string must use: The SMS default 7-bit coded alphabet as defined in 3GPP TS 23.038 with bit 8 set to 9.

8.173.2 Field Documentation

8.173.2.1 uint8_t nas_serviceProviderName::displayCondition

8.173.2.2 uint8_t nas_serviceProviderName::spn[255]

8.173.2.3 uint8_t nas_serviceProviderName::spnLength

8.174 nas_servSystem Struct Reference

Data Fields

- uint8_t regState
- uint8_t csAttachState
- uint8_t psAttachState
- uint8_t selNetwork
- uint8_t numRadioInterfaces
- uint8_t radioInterface [32]

8.174.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

regState	• Registration state - Registration state of the mobile • Values: – 0 - Not Registered; mobile is not currently searching for a new network to provide service – 1 - Registered with a network – 2 - Not registered, but mobile is currently searching for a new network to provide service – 3 - Registration denied by visible network – 4 - Registration state is unknown

csAttachState	• CS Attach State - Circuit-switched domain attach state of the mobile • Values: – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached
psAttachState	• PS Attach State - Packet-switched domain attach state of the mobile • Values: – 0 - Unknown or not applicable – 1 - Attached – 2 - Detached
selNetwork	• Selected Network - Type of selected radio access network • Values: – 0 - Unknown – 1 - 3GPP2 network – 2 - 3GPP network
numRadio-Interfaces	• In Use Radio Interfaces Number – Number of radio interfaces currently in use – defaults to zero
radioInterface	• Radio Interface currently in use • Values: – 0x00 - RADIO_IF_NO_SVC - None(no service) – 0x01 - RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x03 - RADIO_IF_AMPS - AMPS – 0x04 - RADIO_IF_GSM - GSM – 0x05 - RADIO_IF_UMTS - UMTS – 0x08 - RADIO_IF_LTE - LTE

8.174.2 Field Documentation

8.174.2.1 uint8_t nas_servSystem::csAttachState

8.174.2.2 uint8_t nas_servSystem::numRadioInterfaces

8.174.2.3 uint8_t nas_servSystem::psAttachState

8.174.2.4 uint8_t nas_servSystem::radioInterface[32]

8.174.2.5 uint8_t nas_servSystem::regState

8.174.2.6 uint8_t nas_servSystem::selNetwork

8.175 nas_sidNid Struct Reference

Data Fields

- uint16_t nid
- uint16_t sid

8.175.1 Detailed Description

This structure contains the parameters for SidNid Information

Parameters

nid	• Network ID
sid	• System ID

8.175.2 Field Documentation

8.175.2.1 uint16_t nas_sidNid::nid

8.175.2.2 uint16_t nas_sidNid::sid

8.176 nas_SignalStrengthTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- int8_t signalStrength
- uint32_t radioInterface

8.176.1 Detailed Description

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
signalStrength	signal strength
radioInterface	radio interface technology of the signal being measured

8.176.2 Field Documentation

8.176.2.1 uint32_t nas_SignalStrengthTlv::radioInterface

8.176.2.2 int8_t nas_SignalStrengthTlv::signalStrength

8.176.2.3 uint8_t nas_SignalStrengthTlv::TlvPresent

8.177 nas_SLQSSignalStrengthsIndReq Struct Reference

Data Fields

- uint8_t rxSignalStrengthDelta
- uint8_t ecioDelta
- uint8_t ioDelta
- uint8_t sinrDelta
- uint8_t rsrqDelta
- uint8_t ecioThresholdListLen
- int16_t ecioThresholdList [10]
- uint8_t sinrThresholdListLen
- uint8_t sinrThresholdList [5]
- uint16_t lteSnrDelta
- uint8_t lteRsrpDelta

8.177.1 Detailed Description

Structure for storing the input parameters passed for SLQSSetSignalStrengthsCallback by the user.

Parameters

rxSignalStrengthDelta	• RSSI delta(in dBm) at which an event report indication, including the current RSSI, will be sent to the requesting control point.
ecioDelta	• ECIO delta at which an event report indication, ecioDelta including the current ECIO, will be sent to the requesting control point. • ECIO delta is an unsigned 1 byte value that increments in negative 0.5 dBm, e.g., ecio_delta of 2 means a change of -1 dBm.
ioDelta	• IO delta (in dBm) at which an event report indication, ioDelta including the current IO, will be sent to the requesting control point.
sinrDelta	• SINR delta level at which an event report indication, sinrDelta including the current SINR, will be sent to the requesting control point.
rsrqDelta	• RSRQ delta level at which an event report indication, including the current RSRQ, will be sent to the requesting control point.
ecioThresholdListLen	• Number of elements in the ECIO threshold list.
ecioThresholdList	• A sequence of thresholds delimiting EcIo event reporting bands. Every time a new EcIo value crosses a threshold value, an event report indication message with the new ECIO value is sent to the requesting control point. For this field: – Maximum number of threshold values is 10 – At least one value must be specified.
sinrThresholdListLen	• Number of elements in the SINR threshold list.

sinrThreshold-List	• A sequence of thresholds delimiting SINR event reporting bands. Every time a new SINR value crosses a threshold value, an event report indication message with the new sinr value is sent to the requesting control point. For this field: – Maximum number of threshold values is 5 – At least one value must be specified.
ltesnrdelta	• LTE SNR delta level at which an event report indication, including the current SNR, will be sent to the requesting control point. LTE SNR delta level is an unsigned 2 byte value, representing the delta in units of 0.1 dB, e.g., lte_snr_delta of 3 means a change 0.3dB.
ltersrpdelta	-LTE RSRP delta level at which an event report -indication, including the current RSRP, will be sent -to the requesting control point. LTE RSRP delta -level is an unsigned 1 byte value, representing the -delta in dB.

8.177.2 Field Documentation

8.177.2.1 uint8_t nas_SLQSSignalStrengthsIndReq::ecioDelta

8.177.2.2 int16_t nas_SLQSSignalStrengthsIndReq::ecioThresholdList[10]

8.177.2.3 uint8_t nas_SLQSSignalStrengthsIndReq::ecioThresholdListLen

8.177.2.4 uint8_t nas_SLQSSignalStrengthsIndReq::ioDelta

8.177.2.5 uint8_t nas_SLQSSignalStrengthsIndReq::lteRsrpDelta

8.177.2.6 uint16_t nas_SLQSSignalStrengthsIndReq::lteSnrDelta

8.177.2.7 uint8_t nas_SLQSSignalStrengthsIndReq::rsrqDelta

8.177.2.8 uint8_t nas_SLQSSignalStrengthsIndReq::rxSignalStrengthDelta

8.177.2.9 uint8_t nas_SLQSSignalStrengthsIndReq::sinrDelta

8.177.2.10 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdList[5]

8.177.2.11 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdListLen

8.178 nas_SLQSSignalStrengthsInformation Struct Reference

Data Fields

- nas_rxSignalStrengthListElement rxSignalStrengthInfo
- nas_ecioListElement ecioInfo
- uint32_t io
- uint8_t sinr
- nas_errorRateListElement errorRateInfo
- nas_rsrqInformation rsrqInfo
- nas_lteSnrinformation lteSnrinfo
- nas_lteRsrpinformation lteRsrpinfo

8.178.1 Detailed Description

Parameters

rxSignal-StrengthInfo	signal strength info list
ecioInfo	ecio info list
io	received IO in dBm; IO is only applicable for 1xEV-DO
sinr	SINR level
errorRateInfo	error rate info
rsrqInfo	rsrq info
lteSnrinfo	lte Snr information
lteRsrpinfo	lte rsrp info

8.178.2 Field Documentation

8.178.2.1 nas_ecioListElement nas_SLQSSignalStrengthsInformation::ecioInfo

8.178.2.2 nas_errorRateListElement nas_SLQSSignalStrengthsInformation::errorRateInfo

8.178.2.3 uint32_t nas_SLQSSignalStrengthsInformation::io

8.178.2.4 nas_lteRsrpinformation nas_SLQSSignalStrengthsInformation::lteRsrpinfo

8.178.2.5 nas_lteSnrinformation nas_SLQSSignalStrengthsInformation::lteSnrinfo

8.178.2.6 nas_rsrqInformation nas_SLQSSignalStrengthsInformation::rsrqInfo

8.178.2.7 nas_rxSignalStrengthListElement nas_SLQSSignalStrengthsInformation::rxSignalStrengthInfo

8.178.2.8 uint8_t nas_SLQSSignalStrengthsInformation::sinr

8.179 nas_SLQSSignalStrengthsTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- nas_SLQSSignalStrengthsInformation sSLQSSignalStrengthsInfo

8.179.1 Detailed Description

Parameters

TlvPresent	indicating the presence of the TLV in the QMI ind
sSLQSSignal-StrengthsInfo	signal strength info

8.179.2 Field Documentation

8.179.2.1 nas_SLQSSignalStrengthsInformation nas_SLQSSignalStrengthsTlv::sSLQSSignalStrengthsInfo

8.179.2.2 uint8_t nas_SLQSSignalStrengthsTlv::TlvPresent

8.180 nas_SrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t isPrefDataPath

8.180.1 Detailed Description

Structure for storing the service status information for CDMA and HDR networks.

Parameters

srvStatus	• Service status of the system. – 0x00 - No service – 0x01 - Limited service – 0x02 - Service – 0x03 - Limited regional service – 0x04 - Power save – 0xFF - Not Available
isPrefDataPath	• Whether the RAT is the preferred data path. – 0x00 - Not preferred – 0x01 - Preferred – 0xFF - Not Available

8.180.2 Field Documentation

8.180.2.1 uint8_t nas_SrvStatusInfo::isPrefDataPath

8.180.2.2 uint8_t nas_SrvStatusInfo::srvStatus

8.181 nas_sysInfoCommon Struct Reference

Data Fields

- uint8_t srvDomainValid
- uint8_t srvDomain
- uint8_t srvCapabilityValid
- uint8_t srvCapability
- uint8_t roamStatusValid
- uint8_t roamStatus
- uint8_t isSysForbiddenValid
- uint8_t isSysForbidden

8.181.1 Detailed Description

Structure for storing the System Information common to CDMA, HDR, GSM, WCDMA and LTE networks.

Parameters

srvDomainValid	• Indicates whether the service domain is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
srvDomain	• Service domain registered on the system. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
srvCapability-Valid	• Indicates whether the service capability is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
srvCapability	• Current system's service capability. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
roamStatusValid	• Indicates whether the roaming status is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

roamStatus	• Current roaming status. – 0x00 - Off – 0x01 - On – 0x02 - Blinking – 0x03 - Out of the neighborhood – 0x04 - Out of the building – 0x05 - Preferred system – 0x06 - Available system – 0x07 - Alliance partner – 0x08 - Premium partner – 0x09 - Full service – 0x0A - Partial service – 0x0B - Banner is on – 0x0C - Banner is off – 0x0D to 0x3F - Reserved for Standard Enhanced Roaming Indicator Numbers – 0x40 to 0x7F - Reserved for Non-Standard Enhanced Roaming Indicator Numbers – 0x40 to 0xFF - Reserved. – 0xFF - Not Available • Values from 0x02 onward are only applicable for 3GPP2
isSysForbidden-Valid	• Indicates whether the forbidden system is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
isSysForbidden	• Whether the system is forbidden. – 0x00 - Not forbidden – 0x01 - Forbidden – 0xFF - Not Available

8.181.2 Field Documentation

8.181.2.1 uint8_t nas_sysInfoCommon::isSysForbidden

8.181.2.2 uint8_t nas_sysInfoCommon::isSysForbiddenValid

8.181.2.3 uint8_t nas_sysInfoCommon::roamStatus

8.181.2.4 uint8_t nas_sysInfoCommon::roamStatusValid

8.181.2.5 uint8_t nas_sysInfoCommon::srvCapability

8.181.2.6 uint8_t nas_sysInfoCommon::srvCapabilityValid

8.181.2.7 uint8_t nas_sysInfoCommon::srvDomain

8.181.2.8 uint8_t nas_sysInfoCommon::srvDomainValid

8.182 nas_TDSCDMAECIOThresh Struct Reference

Data Fields

- uint8_t TDSCDMAECIOThreshListLen
- float * pTDSCDMAECIOThreshList

8.182.1 Detailed Description

This structure contains TDSCDMA ECIO threshold related parameters.

Parameters

TDSCDMAECIOThreshListLen	• Length of the TDSCDMA ECIO threshold list parameter to follow
pTDSCDMAECIOThreshList	• Array of ECIO thresholds (in dB) used by TD-SCDMA • Maximum of 32 values.

8.182.2 Field Documentation

8.182.2.1 float* nas_TDSCDMAECIOThresh::pTDSCDMAECIOThreshList

8.182.2.2 uint8_t nas_TDSCDMAECIOThresh::TDSCDMAECIOThreshListLen

8.183 nas_TDSCDMARSCPThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSCPThreshListLen
- int16_t * pTDSCDMARSCPThreshList

8.183.1 Detailed Description

This structure contains TDSCDMA RSCP threshold related parameters.

Parameters

TDSCDMARSCPThreshListLen	• Length of the TDSCDMA RSCP threshold list parameter to follow
pTDSCDMARSCPThreshList	• Array of RSCP thresholds (in units of 0.1 dBm) • Maximum of 32 values • Range for RSCP values: -120 to -25 (in dBm).

8.183.2 Field Documentation

8.183.2.1 int16_t* nas_TDSCDMARSCPThresh::pTDSCDMARSCPThreshList

8.183.2.2 uint8_t nas_TDSCDMARSCPThresh::TDSCDMARSCPThreshListLen

8.184 nas_TDSCDMARSSIThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSSIThreshListLen
- float * pTDSCDMARSSIThreshList

8.184.1 Detailed Description

This structure contains TDSCDMA RSSI threshold related parameters.

Parameters

TDSCDMARSSIThreshListLen	• Length of the TDSCDMA RSSI threshold list parameter to follow
pTDSCDMARSSIThreshList	• Array of RSSI thresholds (in dBm) used by TD-SCDMA • Maximum of 32 values.

8.184.2 Field Documentation

8.184.2.1 float* nas_TDSCDMARSSIThresh::pTDSCDMARSSIThreshList

8.184.2.2 uint8_t nas_TDSCDMARSSIThresh::TDSCDMARSSIThreshListLen

8.185 nas_TDSCDMASINRCONFThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRCONFThreshListLen
- float * pTDSCDMASINRCONFThreshList

8.185.1 Detailed Description

Parameters

TDSCDMASINRCONFThreshListLen	• Length of the TDSCDMA SINR threshold list parameter to follow
pTDSCDMASINRCONFThreshList	• Array of SINR thresholds (in dB) used by TD-SCDMA • Maximum of 32 values

8.185.2 Field Documentation

8.185.2.1 float* nas_TDSCDMASINRCONFThresh::pTDSCDMASINRCONFThreshList

8.185.2.2 uint8_t nas_TDSCDMASINRCONFThresh::TDSCDMASINRCONFThreshListLen

8.186 nas_TDSCDMASINRThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRThreshListLen
- float * pTDSCDMASINRThreshList

8.186.1 Detailed Description

This structure contains TDSCDMA SINR threshold related parameters.

Parameters

TDSCDMASINRThreshListLen	• Length of the TDSCDMA SINR threshold list parameter to follow
pTDSCDMASINRThreshList	• Array of SINR thresholds (in dB) used by TD-SCDMA • Maximum of 32 values

8.186.2 Field Documentation

8.186.2.1 float* nas_TDSCDMASINRThresh::pTDSCDMASINRThreshList

8.186.2.2 uint8_t nas_TDSCDMASINRThresh::TDSCDMASINRThreshListLen

8.187 nas_timeInfo Struct Reference

Data Fields

- uint16_t year
- uint8_t month
- uint8_t day
- uint8_t hour
- uint8_t minute
- uint8_t second
- uint8_t dayOfWeek
- int8_t timeZone
- uint8_t dayLtSavingAdj
- uint8_t radioInterface
- uint8_t TlvPresent

8.187.1 Detailed Description

This structure contains the parameters for Network Time.

Parameters

year	• Year
month	• Month • 1 is January and 12 is December
day	• Day • Range - 1 to 31
hour	• Hour • Range - 0 to 59
minute	• Minute • Range - 0 to 59
second	• Second • Range - 0 to 59
dayOfWeek	• Day of the week • 0 is Monday and 6 is Sunday
timeZone	• Offset from Universal time • The difference between local time and Universal time, in increments of 15 min • Signed Value
dayLtSavingAdj	• Daylight saving adjustment in hours • Possible values - 0, 1, and 2. • This field is ignored if radio_if is NAS_RADIO_IF_CDMA_1XEVDO
radioInterface	• Radio interface from which the information comes • Values – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA -TD-SCDMA
TlvPresent	• Tlv Present.

8.187.2 Field Documentation

8.187.2.1 uint8_t nas_timeInfo::day

8.187.2.2 uint8_t nas_timeInfo::dayLtSavingAdj

8.187.2.3 uint8_t nas_timeInfo::dayOfWeek

8.187.2.4 uint8_t nas_timeInfo::hour

8.187.2.5 uint8_t nas_timeInfo::minute

8.187.2.6 uint8_t nas_timeInfo::month

8.187.2.7 uint8_t nas_timeInfo::radioInterface

8.187.2.8 uint8_t nas_timeInfo::second

8.187.2.9 int8_t nas_timeInfo::timeZone

8.187.2.10 uint8_t nas_timeInfo::TlvPresent

8.187.2.11 uint16_t nas_timeInfo::year

8.188 nas_trueIMSI Struct Reference

Data Fields

- uint8_t mccT [3]
- uint16_t imsiT1112
- uint8_t imsiTS1 [7]
- uint8_t imsiTS2 [3]
- uint8_t imsiTaddrNum

8.188.1 Detailed Description

Parameters

mccT	• ASCII character representation of MCC_T
imsiT1112	• ASCII character representation of IMSI_T_11_12 value – 0xFFFF - Not Available
imsiTS1	• ASCII character representation of IMSI_T_S1 value
imsiTS2	• ASCII character representation of IMSI_T_S2 value
imsiTaddrNum	• Value of IMSI_T_ADDR_NUM – 0xFF - Not Available

8.188.2 Field Documentation

8.188.2.1 uint16_t nas_truelMSI::imsiT1112

8.188.2.2 uint8_t nas_truelMSI::imsiTaddrNum

8.188.2.3 uint8_t nas_truelMSI::imsiTS1[7]

8.188.2.4 uint8_t nas_truelMSI::imsiTS2[3]

8.188.2.5 uint8_t nas_truelMSI::mccT[3]

8.189 nas_txInfo Struct Reference

Data Fields

- uint8_t isInTraffic
- int32_t txPower

8.189.1 Detailed Description

Parameters

isInTraffic	• Whether the device is in traffic. – 0x00 - not in traffic – 0x01 - in traffic • The txPower field is only meaningful when in the device is in traffic.
txPower	• Tx power value in 1/10 dbm.

8.189.2 Field Documentation

8.189.2.1 uint8_t nas_txInfo::isInTraffic

8.189.2.2 int32_t nas_txInfo::txPower

8.190 nas_UMTSInfo Struct Reference

Data Fields

- uint16_t cellID
- uint8_t plmn [3]
- uint16_t lac
- uint16_t uarfcn
- uint16_t psc
- int16_t rscp
- int16_t ecio
- uint8_t umtsInst
- nas_UMTSinstInfo UMTSInstInfo [255]
- uint8_t geranInst
- nas_geranInstInfo GeranInstInfo [255]

8.190.1 Detailed Description

This structure contains information about the UMTS Network.

Parameters

cellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
plmn[NAS_PLMN_LENGTH]	• MCC/MNC information coded as octet 3, 4, and 5. • This field is ignored when nmrCellID is not present.
lac	• Location area code. • This field is ignored when nmrCellID is not present. – 0xFFFF - Not Available
uarfcn	• UTRA absolute RF channel number. – 0xFFFF - Not Available
psc	• Primary scrambling code. – 0xFFFF - Not Available
rscp	• Received signal code power. – 0xFFFF - Not Available
ecio	• ECIO(Signal-to-Interference-ratio). – 0xFFFF - Not Available
umtsInst	• Provides the number of set of UMTS info instances. • If 0(zero), then no information follows it.
UMTSInstInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_UMTSinstInfo for more information.
geranInst	• Provides the number of set of GERAN info instances. • If 0(zero), then no information follows it.
GeranInstInfo[NAS_MAX_DESCRIPTION_LENGTH]	• See nas_geranInstInfo for more information.

8.190.2 Field Documentation

8.190.2.1 uint16_t nas_UMTSInfo::cellID

8.190.2.2 int16_t nas_UMTSInfo::ecio

8.190.2.3 uint8_t nas_UMTSInfo::geranInst

8.190.2.4 nas_geranInstInfo nas_UMTSInfo::GeranInstInfo[255]

8.190.2.5 uint16_t nas_UMTSInfo::lac

8.190.2.6 uint8_t nas_UMTSInfo::plmn[3]

8.190.2.7 uint16_t nas_UMTSInfo::psc

8.190.2.8 int16_t nas_UMTSInfo::rscp

8.190.2.9 uint16_t nas_UMTSInfo::uarfcn

8.190.2.10 uint8_t nas_UMTSInfo::umtsInst

8.190.2.11 nas_UMTSinstInfo nas_UMTSInfo::UMTSInstInfo[255]

8.191 nas_UMTSinstInfo Struct Reference

Data Fields

- uint16_t umtsUarfcn
- uint16_t umtsPsc
- int16_t umtsRscp
- int16_t umtsEcio

8.191.1 Detailed Description

This structure contains information about the UMTS Instances in UMTS Network.

Parameters

umtsUarfcn	• UTRA absolute RF channel number.
umtsPsc	• Primary scrambling code.
umtsRscp	• Received signal code power.
umtsEcio	• ECIO(Signal-to-Interference-ratio).

8.191.2 Field Documentation

8.191.2.1 int16_t nas_UMTSinstInfo::umtsEcio

8.191.2.2 uint16_t nas_UMTSinstInfo::umtsPsc

8.191.2.3 int16_t nas_UMTSinstInfo::umtsRscp

8.191.2.4 uint16_t nas_UMTSinstInfo::umtsUarfcn

8.192 nas_umtsLTENbrCell Struct Reference

Data Fields

- uint16_t earfcn
- uint16_t pci
- uint32_t rsrp
- uint32_t rsrq
- int16_t srxlev
- uint8_t cellIsTDD

8.192.1 Detailed Description

This structure contains information about the UMTS LTE neighbour Cell.

Parameters

earfcn	• E-UTRA absolute RF channel number of the detected cell.
pci	• Physical cell ID of the detected cell. • Range is defined in 3GPP TS 36.211
rsrp	• Current received signal strength indication (in dBm) of the detected cell.
rsrq	• Current reference signal received quality (in dB) of the detected cell.
srxlev	• Cell selection Rx level (Srxlev) value of the detected cell in linear scale. • This field is only valid when wcdma_rrc_state is not NAS_WCDMA_RRC_STATE_CEL_FACH or NAS_WCDMA_RRC_STATE_CELL_DCH.
cellIsTDD	• TRUE if the cell is TDD; FALSE if the cell is FDD.

8.192.2 Field Documentation

8.192.2.1 uint8_t nas_umtsLTENbrCell::cellIsTDD

8.192.2.2 uint16_t nas_umtsLTENbrCell::earfcn

8.192.2.3 uint16_t nas_umtsLTENbrCell::pci

8.192.2.4 uint32_t nas_umtsLTENbrCell::rsrp

8.192.2.5 uint32_t nas_umtsLTENbrCell::rsrq

8.192.2.6 int16_t nas_umtsLTENbrCell::srxlev

8.193 nas_UniversalTime Struct Reference

Data Fields

- uint16_t year
- uint8_t month
- uint8_t day
- uint8_t hour
- uint8_t minute
- uint8_t second
- uint8_t dayOfWeek

8.193.1 Detailed Description

This structure contains the parameters for Universal Time Information.

Parameters

year	• Year.
month	• Month. – 1 is January and 12 is December.
day	• Day. – Range 1 to 31.
hour	• Hour. – Range 0 to 59.
minute	• Minute. – Range 0 to 59.
second	• Second. – Range 0 to 59.
dayOfWeek	• Day of the Week. – 0 is Monday and 6 is Sunday.

8.193.2 Field Documentation

8.193.2.1 uint8_t nas_UniversalTime::day

8.193.2.2 uint8_t nas_UniversalTime::dayOfWeek

8.193.2.3 uint8_t nas_UniversalTime::hour

8.193.2.4 uint8_t nas_UniversalTime::minute

8.193.2.5 uint8_t nas_UniversalTime::month

8.193.2.6 uint8_t nas_UniversalTime::second

8.193.2.7 uint16_t nas_UniversalTime::year

8.194 nas_wcdmaCellInfo Struct Reference

Data Fields

- uint16_t psc
- int16_t cpich_rscp
- int16_t cpich_ecno
- int16_t srxlev

8.194.1 Detailed Description

This structure contains information about the WCDMA Cell.

Parameters

psc	• Primary scrambling code. • Range: 0 to 511.
cpich_rscp	• Absolute power level (in 1/10 dBm) of the common pilot channel as received by the UE. • Range: -120.0 dBm to -25.0 dBm
cpich_ecno	• CPICH Ec/No; ratio (in 1/10 dB) of the received energy per PN chip for the CPICH to the total received power spectral density at the UE antenna connector. • Range: -50.0 dB to 0.
srxlev	• Cell selection Rx level (Srxlev) value. • Range: -128 to 128. • This field is only valid when ue_in_idle is TRUE.

8.194.2 Field Documentation

8.194.2.1 int16_t nas_wcdmaCellInfo::cpich_ecno

8.194.2.2 int16_t nas_wcdmaCellInfo::cpich_rscp

8.194.2.3 uint16_t nas_wcdmaCellInfo::psc

8.194.2.4 int16_t nas_wcdmaCellInfo::srxlev

8.195 nas_WCDMAECIOThresh Struct Reference

Data Fields

- uint8_t WCDMAECIOThreshListLen
- int16_t * pWCDMAECIOThreshList

8.195.1 Detailed Description

This structure contains WCDMA ECIO threshold related parameters.

Parameters

WCDMAECIO-ThreshListLen	• Length of the WCDMA ECIO threshold list parameter to follow
pWCDMAECIO-ThreshList	• Array of ECIO thresholds (in units of 0.1 dB) • Maximum of 32 values • Range for ECIO values: -31.5 to 0 (in dB)

8.195.2 Field Documentation

8.195.2.1 int16_t* nas_WCDMAECIOThresh::pWCDMAECIOThreshList

8.195.2.2 uint8_t nas_WCDMAECIOThresh::WCDMAECIOThreshListLen

8.196 nas_WCDMAInfoLTENeighborCell Struct Reference

Data Fields

- uint32_t wcdmaRRCState
- uint8_t umtsLTENbrCellLen
- nas_umtsLTENbrCell UMTSLTENbrCell [255]

8.196.1 Detailed Description

This structure contains information about the WCDMA - LTE Neighboring Cell Info Set.

Parameters

wcdmaRRC-State	• WCDMA RRC states. • Defined in 3GPP TS 25.331 • Values: – 0x00 - NAS_WCDMA_RRC_STATE_DISCONNECTED * WCDMA RRC State is IDLE – 0x01 - NAS_WCDMA_RRC_STATE_CELL_PCH * WCDMA RRC state is CELL_PCH – 0x02 - NAS_WCDMA_RRC_STATE_URA_PCH * WCDMA RRC state is URA_PCH – 0x03 - NAS_WCDMA_RRC_STATE_CELL_FACH * WCDMA RRC state is CELL_FACH – 0x04 - NAS_WCDMA_RRC_STATE_CELL_DCH * WCDMA RRC state is CELL_DCH
umtsLTENbr-CellLen	• Number of sets of UMTS LTE Neighbors.
UMTSLTENbr-Cell	• See nas_umtsLTENbrCell for more information.

8.196.2 Field Documentation

8.196.2.1 nas_umtsLTENbrCell nas_WCDMAInfoLTENeighborCell::UMTSLTENbrCell[255]

8.196.2.2 uint8_t nas_WCDMAInfoLTENeighborCell::umtsLTENbrCellLen

8.196.2.3 uint32_t nas_WCDMAInfoLTENeighborCell::wcdmaRRCState

8.197 nas_WCDMARSSIThresh Struct Reference

Data Fields

- uint8_t WCDMARSSIThreshListLen
- int16_t * pWCDMARSSIThreshList

8.197.1 Detailed Description

This structure contains WCDMA RSSI threshold related parameters.

Parameters

WCDMARSSI-ThreshListLen	• Length of the WCDMA RSSI threshold list parameter to follow
pWCDMARSSI-ThreshList	• Array of RSSI thresholds (in units of 0.1 dBm) • Maximum of 32 values. • Range for RSSI values: -121 to 0 (in dBm)

8.197.2 Field Documentation

8.197.2.1 int16_t* nas_WCDMARSSIThresh::pWCDMARSSIThreshList

8.197.2.2 uint8_t nas_WCDMARSSIThresh::WCDMARSSIThreshListLen

8.198 nas_WCDMASysInfo Struct Reference

Data Fields

- nas_sysInfoCommon sysInfoWCDMA
- uint8_t lacValid
- uint16_t lac
- uint8_t cellIdValid
- uint32_t cellId
- uint8_t regRejectInfoValid
- uint8_t rejectSrvDomain
- uint8_t rejCause
- uint8_t networkIdValid
- uint8_t MCC [3]
- uint8_t MNC [3]
- uint8_t hsCallStatusValid
- uint8_t hsCallStatus
- uint8_t hsIndValid
- uint8_t hsInd
- uint8_t pscValid
- uint16_t psc

8.198.1 Detailed Description

Structure for storing the WCDMA System Information.

Parameters

sysInfoWCDMA	• See nas_sysInfoCommon for more information.
lacValid	• Indicates whether the location area code is valid.. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
lac	• Location area code. • Only applies to 3GPP. – 0xFFFF - Not Available
cellIdValid	• Indicates whether the cell ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

cellId	• Cell ID. – 0xFFFFFFFF - Not Available
regRejectInfo-Valid	• Indicates whether the registration reject information is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
rejectSrvDomain	• Type of service domain in which the registration is rejected. – 0x00 - SYS_SRV_DOMAIN_NO_SRV - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched – 0x04 - Camped – 0xFF - Not Available
rejCause	• Reject cause values sent are specified in [3GPP TS 24.008, Section 10.5.3.6]. – 0xFF - Not Available
networkIdValid	• Indicates whether the network ID is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
MCC[NAS_PLM-N_LENGTH]	• Mobile Country Code. • MCC digits in ASCII characters
MNC[NAS_PLM-N_LENGTH]	• Mobile Network Code. • MNC digits in ASCII characters • An unused byte is set to 0xFF. • In case of two-digit MNC values, the third (unused) digit is set to 0xFF. For example, 15 (a two-digit MNC) is reported using the byte stream 0x31 0x35 0xFF.
hsCallStatus-Valid	• Indicates whether the high-speed call status is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available

hsCallStatus	• Call status on high speed. • Only applicable for WCDMA. – 0x00 - HSDPA and HSUPA are unsupported – 0x01 - HSDPA is supported – 0x02 - HSUPA is supported – 0x03 - HSDPA and HSUPA are supported – 0x04 - HSDPA+ is supported – 0x05 - HSDPA+ and HSUPA are supported – 0x06 - Dual-cell HSDPA+ is supported – 0x07 - Dual-cell HSDPA+ and HSUPA are supported – 0xFF - Not Available
hsIndValid	• Indicates whether high-speed service indication is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
hsInd	• High-speed service indication • Only applicable for WCDMA. – 0x00 - HSDPA and HSUPA are unsupported – 0x01 - HSDPA is supported – 0x02 - HSUPA is supported – 0x03 - HSDPA and HSUPA are supported – 0x04 - HSDPA+ is supported – 0x05 - HSDPA+ and HSUPA are supported – 0x06 - Dual-cell HSDPA+ is supported – 0x07 - Dual-cell HSDPA+ and HSUPA are supported – 0xFF - Not Available
pscValid	• Indicates whether primary scrambling code is valid. – 0x00 - Invalid – 0x01 - Valid – 0xFF - Not Available
psc	• Primary scrambling code. – 0xFFFF - Not Available

8.198.2 Field Documentation

8.198.2.1 uint32_t nas_WCDMASysInfo::cellId

8.198.2.2 uint8_t nas_WCDMASysInfo::cellIdValid

8.198.2.3 uint8_t nas_WCDMASysInfo::hsCallStatus

8.198.2.4 uint8_t nas_WCDMASysInfo::hsCallStatusValid

8.198.2.5 uint8_t nas_WCDMASysInfo::hsInd

8.198.2.6 uint8_t nas_WCDMASysInfo::hsIndValid

8.198.2.7 uint16_t nas_WCDMASysInfo::lac

8.198.2.8 uint8_t nas_WCDMASysInfo::lacValid

8.198.2.9 uint8_t nas_WCDMASysInfo::MCC[3]

8.198.2.10 uint8_t nas_WCDMASysInfo::MNC[3]

8.198.2.11 uint8_t nas_WCDMASysInfo::networkIdValid

8.198.2.12 uint16_t nas_WCDMASysInfo::psc

8.198.2.13 uint8_t nas_WCDMASysInfo::pscValid

8.198.2.14 uint8_t nas_WCDMASysInfo::regRejectInfoValid

8.198.2.15 uint8_t nas_WCDMASysInfo::rejCause

8.198.2.16 uint8_t nas_WCDMASysInfo::rejectSrvDomain

8.198.2.17 nas_sysInfoCommon nas_WCDMASysInfo::sysInfoWCDMA

8.199 nas_wcdmaUARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t uarfcn

8.199.1 Detailed Description

Parameters

status	• 0 - Disable • 1 - Enable
uarfcn	• UARFCN to which UE is locked

8.199.2 Field Documentation

8.199.2.1 uint8_t nas_wcdmaUARFCN::status

8.199.2.2 uint32_t nas_wcdmaUARFCN::uarfcn

8.200 NASBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t band_pref

8.200.1 Detailed Description

Structure used to store all Band Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
band_pref	- Roaming Indication Value

8.200.2 Field Documentation

8.200.2.1 uint64_t NASBandPreferenceTlv::band_pref

8.200.2.2 uint8_t NASBandPreferenceTlv::TlvPresent

8.201 NASEmergencyModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EmerMode

8.201.1 Detailed Description

Structure used to store Emergency Mode TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
EmerMode	- Emergency Mode Value

8.201.2 Field Documentation

8.201.2.1 uint8_t NASEmergencyModeTlv::EmerMode

8.201.2.2 uint8_t NASEmergencyModeTlv::TlvPresent

8.202 NasGetLTECphyCaInfo Struct Reference

Data Fields

- NASPhyCaAggScellIndType PhyCaAggScellIndType
- NASPhyCaAggScellDlBw PhyCaAggScellDlBw
- NASPhyCaAggScellInfo PhyCaAggScellInfo
- NASPhyCaAggPcellInfo PhyCaAggPcellInfo

- NASPhyCaAggScellIndex PhyCaAggScellIndex

8.202.1 Field Documentation

8.202.1.1 NASPhyCaAggPcellInfo NasGetLTECphyCaInfo::PhyCaAggPcellInfo

8.202.1.2 NASPhyCaAggScellDlBw NasGetLTECphyCaInfo::PhyCaAggScellDlBw

8.202.1.3 NASPhyCaAggScellIndex NasGetLTECphyCaInfo::PhyCaAggScellIndex

8.202.1.4 NASPhyCaAggScellIndType NasGetLTECphyCaInfo::PhyCaAggScellIndType

8.202.1.5 NASPhyCaAggScellInfo NasGetLTECphyCaInfo::PhyCaAggScellInfo

8.203 NASGWAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t GWAcqOrderPref

8.203.1 Detailed Description

Structure used to store GSM/WCDMA acquisition order preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
GWAcqOrder-Pref	- GSM/WCDMA acquisition order Preference Value

8.203.2 Field Documentation

8.203.2.1 uint32_t NASGWAcqOrderPrefTlv::GWAcqOrderPref

8.203.2.2 uint8_t NASGWAcqOrderPrefTlv::TlvPresent

8.204 NASLTEBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LTEBandPref

8.204.1 Detailed Description

Structure used to store LTE Band Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
LTEBandPref	- LTE Band Preference Value

8.204.2 Field Documentation

8.204.2.1 uint64_t NASLTEBandPreferenceTlv::LTEBandPref

8.204.2.2 uint8_t NASLTEBandPreferenceTlv::TlvPresent

8.205 NASLteNasReleaseInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t nas_release
- uint8_t nas_major
- uint8_t nas_minor

8.205.1 Detailed Description

This structure contains LTE Nas Release Infomation

Parameters

Tlvresult	• unpack result
nas_release	• LTE NAS release
nas_major	• LTE NAS version major
nas_minor	• LTE NAS version minor

8.205.2 Field Documentation

8.205.2.1 uint8_t NASLteNasReleaseInfoTlv::nas_major

8.205.2.2 uint8_t NASLteNasReleaseInfoTlv::nas_minor

8.205.2.3 uint8_t NASLteNasReleaseInfoTlv::nas_release

8.205.2.4 uint8_t NASLteNasReleaseInfoTlv::TlvPresent

8.206 NASModePreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ModePref

8.206.1 Detailed Description

Structure used to store Mode Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
ModePref	- Mode Preference Value

8.206.2 Field Documentation

8.206.2.1 uint16_t NASModePreferenceTlv::ModePref

8.206.2.2 uint8_t NASModePreferenceTlv::TlvPresent

8.207 NASNetSelPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t NetSelPref

8.207.1 Detailed Description

Structure used to store Network Selection Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
NetSelPref	- Network Selection Preference Value

8.207.2 Field Documentation

8.207.2.1 uint8_t NASNetSelPreferenceTlv::NetSelPref

8.207.2.2 uint8_t NASNetSelPreferenceTlv::TlvPresent

8.208 NASOTAMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t message_type
- uint16_t data_len
- uint8_t data_buf [2048]

8.208.1 Detailed Description

This structure contains OTA message

Parameters

Tlvresult	• unpack result
message_type	• message type – 0 - LTE ESM uplink – 1 - LTE ESM downlink – 2 - LTE EMM uplink – 3 - LTE EMM downlink – 4 - GSM/UMTS uplink – 5 - GSM/UMTS downlink
data_len	• OTA Message Content Length
data_buf	• OTA Message Content

8.208.2 Field Documentation

8.208.2.1 uint8_t NASOTAMessageTlv::data_buf[2048]

8.208.2.2 uint16_t NASOTAMessageTlv::data_len

8.208.2.3 uint32_t NASOTAMessageTlv::message_type

8.208.2.4 uint8_t NASOTAMessageTlv::TlvPresent

8.209 NASPhyCaAggPcellInfo Struct Reference

Data Fields

- uint32_t pci
- uint32_t freq
- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint32_t iLTEbandValue
- uint8_t TlvPresent

8.209.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Pcell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.

dl_bw_value	• Downlink Bandwidth Values. • See LITEQMI_NAS_LTE_CPHY_CA_BW_NRB for more information.
scell_state	• Scell state Values. • See LITEQMI_NAS_LTE_CPHY_SCELL_STATE for more information.
TlvPresent	• Tlv Present.

8.209.2 Field Documentation

8.209.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggPcellInfo::dl_bw_value

8.209.2.2 uint32_t NASPhyCaAggPcellInfo::freq

8.209.2.3 uint32_t NASPhyCaAggPcellInfo::iLTEbandValue

8.209.2.4 uint32_t NASPhyCaAggPcellInfo::pci

8.209.2.5 uint8_t NASPhyCaAggPcellInfo::TlvPresent

8.210 NASPhyCaAggScellDlBw Struct Reference

Data Fields

- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint8_t TlvPresent

8.210.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation Downlink Bandwidth of Scell.

Parameters

dl_bw_value	• Downlink Bandwidth Values. • See LITEQMI_NAS_LTE_CPHY_CA_BW_NRB for more information.

8.210.2 Field Documentation

8.210.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellDlBw::dl_bw_value

8.210.2.2 uint8_t NASPhyCaAggScellDlBw::TlvPresent

8.211 NASPhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.211.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Index.

Parameters

scell_idx	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
TlvPresent	• Tlv Present.

8.211.2 Field Documentation

8.211.2.1 uint8_t NASPhyCaAggScellIndex::scell_idx

8.211.2.2 uint8_t NASPhyCaAggScellIndex::TlvPresent

8.212 NASPhyCaAggScellIndType Struct Reference

Data Fields

- uint32_t pci
- uint32_t freq
- LITEQMI_NAS_LTE_CPHY_SCELL_STATE scell_state
- uint8_t TlvPresent

8.212.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Indeicator Type.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
scell_state	• Scell state Values. • See LITEQMI_NAS_LTE_CPHY_SCELL_STATE for more information.

TlvPresent	• Tlv Present.

8.212.2 Field Documentation

8.212.2.1 uint32_t NASPhyCaAggScellIndType::freq

8.212.2.2 uint32_t NASPhyCaAggScellIndType::pci

8.212.2.3 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellIndType::scell_state

8.212.2.4 uint8_t NASPhyCaAggScellIndType::TlvPresent

8.213 NASPhyCaAggScellInfo Struct Reference

Data Fields

- uint32_t pci
- uint32_t freq
- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint32_t iLTEbandValue
- LITEQMI_NAS_LTE_CPHY_SCELL_STATE scell_state
- uint8_t TlvPresent

8.213.1 Detailed Description

This structure contains the parameters for Physical Carrier aggregation of Scell Information.

Parameters

pci	• Physical cell ID of the SCell Range. • Range for ID values: 0 to 503.
freq	• Frequency of the absolute cell Range. • Range for ID values: 0 to 65535.
dl_bw_value	• Downlink Bandwidth Values. • See LITEQMI_NAS_LTE_CPHY_CA_BW_NRB for more information.
iLTEbandValue	• Band value. • Range for LTE Band class 120 to 160.
scell_state	• Scell state Values. • See LITEQMI_NAS_LTE_CPHY_SCELL_STATE for more information.

TlvPresent	• Tlv Present.

8.213.2 Field Documentation

8.213.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellInfo::dl_bw_value

8.213.2.2 uint32_t NASPhyCaAggScellInfo::freq

8.213.2.3 uint32_t NASPhyCaAggScellInfo::iLTEbandValue

8.213.2.4 uint32_t NASPhyCaAggScellInfo::pci

8.213.2.5 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellInfo::scell_state

8.213.2.6 uint8_t NASPhyCaAggScellInfo::TlvPresent

8.214 NASPRLPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t PRLPref

8.214.1 Detailed Description

Structure used to store CDMA PRL Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
PRLPref	- CDMA PRL Preference Value

8.214.2 Field Documentation

8.214.2.1 uint16_t NASPRLPreferenceTlv::PRLPref

8.214.2.2 uint8_t NASPRLPreferenceTlv::TlvPresent

8.215 NASQmiCbkNasSwiOTAMessageInd Struct Reference

Data Fields

- NASOTAMessageTlv otaMsgTlv
- NASLteNasReleaseInfoTlv nasRelInfoTlv
- NASTimeInfoTlv timeTlv

8.215.1 Detailed Description

This structure contains OTA message

Parameters

otaMsgTlv	• See NASOTAMessageTlv for more details
nasRelInfoTlv	• See NASLteNasReleaseInfoTlv for more details
timeTlv	• See NASTimeInfoTlv for more details

8.215.2 Field Documentation

8.215.2.1 NASLteNasReleaseInfoTlv NASQmiCbkNasSwiOTAMessageInd::nasRelInfoTlv

8.215.2.2 NASOTAMessageTlv NASQmiCbkNasSwiOTAMessageInd::otaMsgTlv

8.215.2.3 NASTimeInfoTlv NASQmiCbkNasSwiOTAMessageInd::timeTlv

8.216 NASQmiCbkNasSystemSelPrefInd Struct Reference

Data Fields

- NASEmergencyModeTlv EMTlv
- NASModePreferenceTlv MPTlv
- NASBandPreferenceTlv BPTlv
- NASPRLPreferenceTlv PRLPTlv
- NASRoamPreferenceTlv RPTlv
- NASLTEBandPreferenceTlv LBPTlv
- NASNetSelPreferenceTlv NSPTlv
- NASServDomainPrefTlv SDPTlv
- NASGWAcqOrderPrefTlv GWAOPTlv

8.216.1 Detailed Description

Structure used to store all QMI Notification parameters.

Parameters

EMTlv	- Emergency mode Tlv • See NASEmergencyModeTlv for more details
MPTlv	- Mode preference Tlv • See NASModePreferenceTlv for more details
BPTlv	- Band preference Tlv • See NASBandPreferenceTlv for more details
PRLPTlv	- PRL preference Tlv • See NASPRLPreferenceTlv for more details
RPTlv	- Roaming preference Tlv • See NASRoamPreferenceTlv for more details

LBPTlv	- LTE band preference Tlv • See NASLTEBandPreferenceTlv for more details
NSPTlv	- Network selection preference Tlv • See NASNetSelPreferenceTlv for more details
SDPTlv	- Service domain preference Tlv • See NASServDomainPrefTlv for more details
GWAOPTlv	- GSM WCDMA Acquisition Order Preference Tlv • See NASGWAcqOrderPrefTlv for more details

Note: None

8.216.2 Field Documentation

8.216.2.1 NASBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::BPTlv

8.216.2.2 NASEmergencyModeTlv NASQmiCbkNasSystemSelPrefInd::EMTlv

8.216.2.3 NASGWAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::GWAOPTlv

8.216.2.4 NASLTEBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::LBPTlv

8.216.2.5 NASModePreferenceTlv NASQmiCbkNasSystemSelPrefInd::MPTlv

8.216.2.6 NASNetSelPreferenceTlv NASQmiCbkNasSystemSelPrefInd::NSPTlv

8.216.2.7 NASPRLPreferenceTlv NASQmiCbkNasSystemSelPrefInd::PRLPTlv

8.216.2.8 NASRoamPreferenceTlv NASQmiCbkNasSystemSelPrefInd::RPTlv

8.216.2.9 NASServDomainPrefTlv NASQmiCbkNasSystemSelPrefInd::SDPTlv

8.217 NASRoamPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RoamPref

8.217.1 Detailed Description

Structure used to store Roaming Preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
RoamPref	- Roaming Preference Value

8.217.2 Field Documentation

8.217.2.1 uint16_t NASRoamPreferenceTlv::RoamPref

8.217.2.2 uint8_t NASRoamPreferenceTlv::TlvPresent

8.218 NASServDomainPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SrvDomainPref

8.218.1 Detailed Description

Structure used to store Service domain preference TLV Value.

Parameters

TlvPresent	- Boolean indicating the presence of the TLV in the QMI response
SrvDomainPref	- Service Domain Preference Value

8.218.2 Field Documentation

8.218.2.1 uint32_t NASServDomainPrefTlv::SrvDomainPref

8.218.2.2 uint8_t NASServDomainPrefTlv::TlvPresent

8.219 NASServingSystemInfo Struct Reference

Data Fields

- uint8_t registrationState
- uint8_t csAttachState
- uint8_t psAttachState
- uint8_t selectedNetwork
- uint8_t radioInterfaceNo
- uint8_t radioInterfaceList [255]
- uint8_t hdrPersonality

8.219.1 Detailed Description

This structure will hold the serving system parameters information

Parameters

registrationState	- Registration state of the mobile • 0 - QMI_NAS_NOT_REGISTERED Not registered;mobile is not currently searching for a new network to provide service • 1 - QMI_NAS_REGISTERED Registered with a network • 2 - QMI_NAS_NOT_REGISTERED_SEARCHING Not registered, but mobile is currently searching for a new network to provide service • 3 - QMI_NAS_REGISTRATION_DENIED Registration denied by the visible network • 4 - QMI_NAS_REGISTRATION_UNKNOWN Registration state is unknown

csAttachState	- Circuit Switch domain attach state of the mobile • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached
psAttachState	- Packet domain attach state of the mobile • 0 - Unknown or not applicable • 1 - Attached • 2 - Detached
selectedNetwork	- Type of selected radio access network • 0x00 - Unknown • 0x01 - 3GPP2 network • 0x02 - 3GPP network
radioInterfaceNo	- Number of radio interfaces currently in use; this indicates how many radio_if identifiers follow this field
radioInterface-List	- Radio interface currently in use (each is 1 byte) • 0x00 - None (no service) • 0x01 - cdma2000 1X • 0x02 - cdma2000 HRPD (1xEV-DO) • 0x03 - AMPS • 0x04 - GSM • 0x05 - UMTS • 0x08 - LTE
hdrPersonality	- HDR personality information (valid only for EVDO) • 0x00 - Unknown • 0x01 - HRPD • 0x02 - eHRPD

Note: None

8.219.2 Field Documentation

8.219.2.1 uint8_t NASServingSystemInfo::csAttachState

8.219.2.2 uint8_t NASServingSystemInfo::hdrPersonality

8.219.2.3 uint8_t NASServingSystemInfo::psAttachState

8.219.2.4 uint8_t NASServingSystemInfo::radioInterfaceList[255]

8.219.2.5 uint8_t NASServingSystemInfo::radioInterfaceNo

8.219.2.6 uint8_t NASServingSystemInfo::registrationState

8.219.2.7 uint8_t NASServingSystemInfo::selectedNetwork

8.220 NASTimeInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t time

8.220.1 Detailed Description

This structure contains LTE Nas time Infomation

Parameters

Tlvresult	• unpack result
time	• Seconds in local time since Jan. 6th 1980 00:00:00 UTC

8.220.2 Field Documentation

8.220.2.1 uint64_t NASTimeInfoTlv::time

8.220.2.2 uint8_t NASTimeInfoTlv::TlvPresent

8.221 newMTMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMTMessageInfo MTMessageInfo

8.221.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
MTMessageInfo	• MT Message • See sMSMTMessageInfo for more information

8.221.2 Field Documentation

8.221.2.1 sMSMTMessageInfo newMTMessageTlv::MTMessageInfo

8.221.2.2 uint8_t newMTMessageTlv::TlvPresent

8.222 pack_cat_CATSendEnvelopeCommand_t Struct Reference

Data Fields

- uint32_t cmdID
- uint32_t dataLen
- uint8_t * pData

8.222.1 Detailed Description

Structure used to store envelope command data.

Parameters

cmdID	• Envelope command type – 0x01 - Menu Selection – 0x02 - Event DL User activity – 0x03 - Event DL Idle Screen Available – 0x04 - Event DL Language Selection
dataLen	• Length of pData in bytes
pData	• Encoded envelope data as defined in ETSI TS 102 223, section 7 [Smart Cards: Card Application Toolkit (CAT) – Release 4]

8.222.2 Field Documentation

8.222.2.1 uint32_t pack_cat_CATSendEnvelopeCommand_t::cmdID

8.222.2.2 uint32_t pack_cat_CATSendEnvelopeCommand_t::dataLen

8.222.2.3 uint8_t* pack_cat_CATSendEnvelopeCommand_t::pData

8.223 pack_cat_CATSendTerminalResponse_t Struct Reference

Data Fields

- uint32_t refID
- uint32_t dataLen
- uint8_t * pData

8.223.1 Detailed Description

Structure used to terminal response data.

Parameters

refID	• Proactive command reference ID. The value should be the same as indicated in the CAT event callback data for the relevant proactive command.

dataLen	• Terminal response data length
pData	• Terminal response for the relevant proactive command encoded as per ETSI TS 102 223, section 6.8 [Smart Cards: Card Application Toolkit (CAT) – Release 4]

8.223.2 Field Documentation

8.223.2.1 uint32_t pack_cat_CATSendTerminalResponse_t::dataLen

8.223.2.2 uint8_t* pack_cat_CATSendTerminalResponse_t::pData

8.223.2.3 uint32_t pack_cat_CATSendTerminalResponse_t::refID

8.224 pack_cat_SetCATEventCallback_t Struct Reference

Data Fields

- uint32_t eventMask

8.224.1 Detailed Description

This structure contains sEnables/disables the CAT event callback pack variable.

Parameters

eventMask	• bitmask of CAT events to register for – 0x00000001 - Display Text – 0x00000002 - Get In-Key – 0x00000004 - Get Input – 0x00000008 - Setup Menu – 0x00000010 - Select Item – 0x00000020 - Send SMS - Alpha Identifier – 0x00000040 - Setup Event: User Activity – 0x00000080 - Setup Event: Idle Screen Notify – 0x00000100 - Setup Event: Language Sel Notify – 0x00000200 - Setup Idle Mode Text – 0x00000400 - Language Notification – 0x00000800 - Refresh – 0x00001000 - End Proactive Session

8.224.2 Field Documentation

8.224.2.1 uint32_t pack_cat_SetCATEventCallback_t::eventMask

8.225 pack_dms_ActivateAutomatic_t Struct Reference

Data Fields

- uint8_t actCode [81]

8.225.1 Detailed Description

This structure contains UIM activate automation information

Parameters

actCode	• NULL-terminated string representing activation code (maximum string length of 12). • Specific carrier requirements may dictate actual activation code that is applicable, e.g., "∗22899"

8.225.2 Field Documentation

8.225.2.1 uint8_t pack_dms_ActivateAutomatic_t::actCode[81]

8.226 pack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint8_t list_type
- uint16_t Tlvresult

8.226.1 Detailed Description

This structure contains which customization id or the list type want to retrieve from modem. This TLV is only applicable for 9x30 modules so far

Parameters

cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.
list_type	• list type requested
Tlvresult	• Pack Result

8.226.2 Field Documentation

8.226.2.1 uint8_t pack_dms_GetCustFeaturesV2_t::cust_id[64+1]

8.226.2.2 uint8_t pack_dms_GetCustFeaturesV2_t::list_type

8.226.2.3 uint16_t pack_dms_GetCustFeaturesV2_t::Tlvresult

8.227 pack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint8_t spc [6]

8.227.1 Detailed Description

This structure contains UIM reset to factory default information

Parameters

spc[IN]	• Service programming code in ASCII format (digits 0 to 9 only)

8.227.2 Field Documentation

8.227.2.1 uint8_t pack_dms_ResetToFactoryDefaults_t::spc[6]

8.228 pack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint8_t activationState

8.228.1 Detailed Description

This structure is used to store Set Service Activation Status callback parameter pack.

Parameters

activationState	• Service activation state. • Values – 0 - Do not report – 1 - Report activation state changes

8.228.2 Field Documentation

8.228.2.1 uint8_t pack_dms_SetActivationStatusCallback_t::activationState

8.229 pack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t crashAction

8.229.1 Detailed Description

This structure is used to store pack_dms_SetCrashAction parameters.

Parameters

crashAction	• 0 - USB Memory Download. Modem will reset after a crash and will stay in USB download mode with only DM port enumerated. • 1 - Reset. Modem will reset and come back in ONLINE mode. Minimal crash data will be available and can be extracted with at!gcdump? AT command • 2 - No Action

8.229.2 Field Documentation

8.229.2.1 uint8_t pack_dms_SetCrashAction_t::crashAction

8.230 pack_dms_SetCustFeature_t Struct Reference

Data Fields

- uint32_t GpsEnable
- uint8_t DisableIMSI
- uint16_t IPFamSupport
- uint8_t RMAutoConnect
- uint8_t GPSSel
- uint8_t SMSSupport
- uint8_t IsVoiceEnabled
- uint8_t DHCPRelayEnabled
- uint8_t GPSLPM

8.230.1 Detailed Description

This structure contains current settings of custom features pack parameters

Parameters

GpsEnable	• describes if GPS is enabled or disabled • values: – 0x00 - GPS is disabled – 0x01 - GPS is enabled
DisableIMSI	• optional 1 byte parameter • describes if IMSI display is enabled or disabled • values: – 0x00 - Allow display of IMSI – 0x01 - Do not display IMSI

IPFamSupport	• optional 2 byte BitMask • bitmask representing the IP families supported • values: – 0x01 - IPv4 – 0x02 - IPv6 – 0x04 - IPv4v6
RMAutoConnect	• optional 1 byte parameter • QMI Mode RM Net Auto Connect Support • values: – 0x00 - Not Supported – 0x01 - Supported
GPSSel	• optional 1 byte parameter • GPS Antenna Select • values: – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied
SMSSupport	• optional 1 byte parameter • SMS support • values: – 0x00 - Not supported – 0x01 - supported • Used to determine whether or not to hide SMS from user
IsVoiceEnabled	• optional 1 byte parameter • Voice support • values: – 0x00 - Enable voice on both AT and QMI interface (default) – 0x01 - Reserved – 0x02 - Disable voice on both AT and QMI interface
DHCPRelay-Enabled	• optional 1 byte parameter • DHCP Relay support • values: – 0x00 - Disable DHCP relay – 0x01 - Enable DHCP relay

GPSLPM	• optional 1 byte parameter • GPSLPM support • values: – 0x00 - Enable GPS in Low Power Mode – 0x01 - Disable GPS in Low Power Mode

8.230.2 Field Documentation

8.230.2.1 uint8_t pack_dms_SetCustFeature_t::DHCPRelayEnabled

8.230.2.2 uint8_t pack_dms_SetCustFeature_t::DisableIMSI

8.230.2.3 uint32_t pack_dms_SetCustFeature_t::GpsEnable

8.230.2.4 uint8_t pack_dms_SetCustFeature_t::GPSLPM

8.230.2.5 uint8_t pack_dms_SetCustFeature_t::GPSSel

8.230.2.6 uint16_t pack_dms_SetCustFeature_t::IPFamSupport

8.230.2.7 uint8_t pack_dms_SetCustFeature_t::IsVoiceEnabled

8.230.2.8 uint8_t pack_dms_SetCustFeature_t::RMAutoConnect

8.230.2.9 uint8_t pack_dms_SetCustFeature_t::SMSSupport

8.231 pack_dms_SetCustFeaturesV2_t Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint16_t value_length
- uint8_t cust_value [8+1]
- uint16_t Tlvresult

8.231.1 Detailed Description

This structure contains customization settings set to modem pack

Parameters

cust_id	• Customization ID (Maximum 64 bytes) • NULL terminated ASCII string.
value_length	• length of cust_value field
cust_value	• Customization Setting Value (Maximum 8 bytes)

Tlvresult	• Pack Result

8.231.2 Field Documentation

8.231.2.1 uint8_t pack_dms_SetCustFeaturesV2_t::cust_id[64+1]

8.231.2.2 uint8_t pack_dms_SetCustFeaturesV2_t::cust_value[8+1]

8.231.2.3 uint16_t pack_dms_SetCustFeaturesV2_t::Tlvresult

8.231.2.4 uint16_t pack_dms_SetCustFeaturesV2_t::value_length

8.232 pack_dms_SetEventReport_t Struct Reference

Data Fields

- uint8_t mode

8.232.1 Detailed Description

This structure is used to store pack_dms_SetEventReport parameters.

Parameters

mode	• Operating Mode • 0 - reset • 1 - set • 2 - no change

8.232.2 Field Documentation

8.232.2.1 uint8_t pack_dms_SetEventReport_t::mode

8.233 pack_dms_SetPower_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t Tlvresult

8.233.1 Detailed Description

Structure to store pack the operating mode of the device.

Parameters

mode	• Selected operating mode – See qaGobiApiTablePowerModes.h for power modes
Tlvresult	• pack Tlv Result.

8.233.2 Field Documentation

8.233.2.1 uint32_t pack_dms_SetPower_t::mode

8.233.2.2 uint16_t pack_dms_SetPower_t::Tlvresult

8.234 pack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint8_t USBComp
- uint16_t Tlvresult

8.234.1 Detailed Description

This structure is used to store USB composition information pack paramters

Parameters

USBComp	• Current USB Composition • Values: – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM

Tlvresult	• pack Tlv Result.

8.234.2 Field Documentation

8.234.2.1 uint16_t pack_dms_SetUSBComp_t::Tlvresult

8.234.2.2 uint8_t pack_dms_SetUSBComp_t::USBComp

8.235 pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint8_t resetInfoInd

8.235.1 Detailed Description

This structure contains set registration state for different indication pack

Parameters

resetInfoInd[IN]	• Values – 0 - Disable – 1 - Enable

8.235.2 Field Documentation

8.235.2.1 uint8_t pack_dms_SLQSDmsSwiIndicationRegister_t::resetInfoInd

8.236 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint8_t clear

8.236.1 Detailed Description

This structure contains pack_dms_SLQSSwiGetCrashInfo parameters

Parameters

clear	• 0 - do not clear crash data after response • 1 - clear crash data after response

8.236.2 Field Documentation

8.236.2.1 uint8_t pack_dms_SLQSSwiGetCrashInfo_t::clear

8.237 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.237.1 Detailed Description

This structure is used to store pack_dms_SLQSSwiSetDyingGaspCfg parameters

Parameters

pDestSMSNum[IN]	• SMS Destination Number as string of 8 bit ASCII Characters Max 20 chars. • Optional parameter.
pDestSMSContent[IN]	• SMS COntent as a string of 8 bit ASCII text characters Max 160 chars. • Optional parameter.

8.237.2 Field Documentation

8.237.2.1 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSContent

8.237.2.2 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSNum

8.238 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- char manString [255]
- char modelString [255]
- char swVerString [255]
- char plasmaIDString [255]

8.238.1 Detailed Description

This structure contains SWI set host device info unpack information

Parameters

manString[IN]	• Host Device Manufacturer Name • Null terminated ASCII String
modelString[IN]	• Host Device Model String • Null terminated ASCII string.

swVerString[IN]	• Host Device Software Version String • Null terminated ASCII string
plasmaIDString[-IN]	• Host Device Plasma ID String • Null terminated alphanumeric ASCII string

8.238.2 Field Documentation

8.238.2.1 char pack_dms_SLQSSwiSetHostDevInfo_t::manString[255]

8.238.2.2 char pack_dms_SLQSSwiSetHostDevInfo_t::modelString[255]

8.238.2.3 char pack_dms_SLQSSwiSetHostDevInfo_t::plasmaIDString[255]

8.238.2.4 char pack_dms_SLQSSwiSetHostDevInfo_t::swVerString[255]

8.239 pack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- char nameString [255]
- char versionString [255]

8.239.1 Detailed Description

This structure contains SWI set host OS info pack information

Parameters

nameString[IN]	Host device manufacture String • NULL terminated ASCII string.
VersionString[IN]	Host device model String • NULL terminated ASCII string.

8.239.2 Field Documentation

8.239.2.1 char pack_dms_SLQSSwiSetOSInfo_t::nameString[255]

8.239.2.2 char pack_dms_SLQSSwiSetOSInfo_t::versionString[255]

8.240 pack_dms_UIMChangePIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t oldValue [255]

- uint8_t newValue [255]

8.240.1 Detailed Description

This structure contains UIM Unblock PIN Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
oldValue[IN]	• Old PIN value of PIN to change
newValue[IN]	• New PIN value of PIN to change

8.240.2 Field Documentation

8.240.2.1 uint8_t pack_dms_UIMChangePIN_t::id

8.240.2.2 uint8_t pack_dms_UIMChangePIN_t::newValue[255]

8.240.2.3 uint8_t pack_dms_UIMChangePIN_t::oldValue[255]

8.241 pack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facility

8.241.1 Detailed Description

This structure contains UIM get control key status information

Parameters

facility[IN]	• MT or network facility – 0 - Network Personalization (PN) – 1 - Network Subset Personalization (PU) – 2 - Service Provider Personalization (PP) – 3 - Corporate Personalization (PC) – 4 - UIM Personalization (PF)

8.241.2 Field Documentation

8.241.2.1 uint8_t pack_dms_UIMGetControlKeyStatus_t::facility

8.242 pack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.242.1 Detailed Description

This structure contains UIM Get ICCID pack

Parameters

Tlvresult	• Pack result.

8.242.2 Field Documentation

8.242.2.1 uint16_t pack_dms_UIMGetICCID_t::Tlvresult

8.243 pack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t facility
- uint8_t facilityState
- uint8_t facilityCk [8]

8.243.1 Detailed Description

This structure contains UIM Set control key protection information

Parameters

facility[IN]	• MT or network facility – 0 - Network Personalization (PN) – 1 - Network Subset Personalization (PU) – 2 - Service Provider Personalization (PP) – 3 - Corporate Personalization (PC) – 4 - UIM Personalization (PF)
facilityState[IN]	• UIM facility state – 0 - Deactivated – 1 - Activated – 2 - Blocked
facilityCk[IN]	• Facility depersonalization control key, string in ASCII text (maximum 8 bytes)

8.243.2 Field Documentation

8.243.2.1 uint8_t pack_dms_UIMSetControlKeyProtection_t::facility

8.243.2.2 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityCk[8]

8.243.2.3 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityState

8.244 pack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t id
- uint8_t bEnable
- uint8_t value [255]

8.244.1 Detailed Description

This structure contains PIN Protection Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
bEnable[IN]	• Enable/disable PIN protection, 0 = Disable
pValue[IN]	• PIN value of the PIN to be enabled/disabled

8.244.2 Field Documentation

8.244.2.1 uint8_t pack_dms_UIMSetPINProtection_t::bEnable

8.244.2.2 uint8_t pack_dms_UIMSetPINProtection_t::id

8.244.2.3 uint8_t pack_dms_UIMSetPINProtection_t::value[255]

8.245 pack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t facility
- uint8_t facilityCk [8]

8.245.1 Detailed Description

This structure contains UIM unblock Control Key information

Parameters

facility[IN]	• MT or network facility – 0 - Network Personalization (PN) – 1 - Network Subset Personalization (PU) – 2 - Service Provider Personalization (PP) – 3 - Corporate Personalization (PC) – 4 - UIM Personalization (PF)
facilityCk[IN]	• Facility depersonalization control key, string in ASCII text (maximum 8 bytes)

8.245.2 Field Documentation

8.245.2.1 uint8_t pack_dms_UIMUnblockControlKey_t::facility

8.245.2.2 uint8_t pack_dms_UIMUnblockControlKey_t::facilityCk[8]

8.246 pack_dms_UIMUnblockPIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t pukValue [255]
- uint8_t newPin [255]

8.246.1 Detailed Description

This structure contains UIM Unblock PIN Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
pukValue[IN]	• PUK value of PIN to be unblocked
newPin[IN]	• New PIN value for the PIN to be unblocked

8.246.2 Field Documentation

8.246.2.1 uint8_t pack_dms_UIMUnblockPIN_t::id

8.246.2.2 uint8_t pack_dms_UIMUnblockPIN_t::newPin[255]

8.246.2.3 uint8_t pack_dms_UIMUnblockPIN_t::pukValue[255]

8.247 pack_dms_UIMVerifyPIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t value [255]

8.247.1 Detailed Description

This structure contains PIN Value Information

Parameters

id[IN]	• PIN ID – 1 (PIN1 / CHV1) – 2 (PIN2 / CHV2)
pValue[IN]	• PIN value of the PIN to be enabled/disabled

8.247.2 Field Documentation

8.247.2.1 uint8_t pack_dms_UIMVerifyPIN_t::id

8.247.2.2 uint8_t pack_dms_UIMVerifyPIN_t::value[255]

8.248 pack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.248.1 Detailed Description

This structure contains the Get Image Preference information pack

Parameters

Tlvresult	• Pack result

8.248.2 Field Documentation

8.248.2.1 uint16_t pack_fms_GetImagesPreference_t::Tlvresult

8.249 pack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint16_t TlvresultTlvresult

8.249.1 Detailed Description

This structure contains the Get Stored Images pack

Parameters

Tlvresult	• Pack result

8.249.2 Field Documentation

8.249.2.1 uint16_t pack_fms_GetStoredImages_t::Tlvresult

8.250 pack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t imageListSize
- FMSPrefImageList * pImageList
- uint32_t bForceDownload
- uint8_t modemindex
- uint16_t Tlvresult

8.250.1 Detailed Description

This structure contains the Set Images Preference pack

Parameters

imageListSize	• Image List Size
pImageList	• Image List • See FMSPrefImageList
bForceDownload	• 0 - Not Force Donwload. • 1 - Focrce Download.
modemindex	• Modem Index.
Tlvresult	• Unpack result

8.250.2 Field Documentation

8.250.2.1 uint32_t pack_fms_SetImagesPreference_t::bForceDownload

8.250.2.2 uint32_t pack_fms_SetImagesPreference_t::imageListSize

8.250.2.3 uint8_t pack_fms_SetImagesPreference_t::modemindex

8.250.2.4 FMSPrefImageList* pack_fms_SetImagesPreference_t::pImageList

8.250.2.5 uint16_t pack_fms_SetImagesPreference_t::Tlvresult

8.251 pack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- loc_SVInfo * pSVInfo
- loc_GnssData * pGnssData
- loc_CellDb * pCellDb
- loc_ClkInfo * pClkInfo
- loc_BdsSVInfo * pBdsSVInfo
- uint16_t Tlvresult

8.251.1 Detailed Description

This structure contains LOC delete assist data pack

Parameters

pSVInfo	• Pointer to struct loc_SVInfo. See loc_SVInfo for more information
pGnssData	• Pointer to struct loc_GnssData. See loc_GnssData for more information
pCellDb	• Pointer to struct loc_CellDb. See loc_CellDb for more information
pClkInfo	• Pointer to struct loc_ClkInfo. See loc_ClkInfo for more information
pBdsSVInfo	• Pointer to struct loc_BdsSVInfo. See loc_BdsSVInfo for more information
Tlvresult	• Pack delete assist data request result.

8.251.2 Field Documentation

8.251.2.1 loc_BdsSVInfo* pack_loc_Delete_Assist_Data_t::pBdsSVInfo

8.251.2.2 loc_CellDb* pack_loc_Delete_Assist_Data_t::pCellDb

8.251.2.3 loc_ClkInfo* pack_loc_Delete_Assist_Data_t::pClkInfo

8.251.2.4 loc_GnssData* pack_loc_Delete_Assist_Data_t::pGnssData

8.251.2.5 loc_SVInfo* pack_loc_Delete_Assist_Data_t::pSVInfo

8.251.2.6 uint16_t pack_loc_Delete_Assist_Data_t::Tlvresult

8.252 pack_loc_EventRegister_t Struct Reference

Data Fields

- uint64_t eventRegister
- uint16_t Tlvresult

8.252.1 Detailed Description

This structure contains the Parameter for RegisterEvents

Parameters

eventRegister	• Specifies the events that the control point is interested in receiving. -Values – 0x00000001 - to receive position report event indications – 0x00000002 - to receive satellite report event indications.These reports are sent at a 1 Hz rate. – 0x00000004 - to receive NMEA reports for position and satellites in view.The report is at a 1 Hz rate. – 0x00000008 - to receive NI Notify/Verify request event indications – 0x00000010 - to receive time injection request event indications. – 0x00000020 - to receive predicted orbits request event indications. – 0x00000040 - to receive position injection request event indications. – 0x00000080 - to receive engine state report event indications. – 0x00000100 - to receive fix session status report event indications. – 0x00000200 - to receive Wi-Fi position request event indications. – 0x00000400 - to receive notifications from the location engine indicating its readiness to accept data from the sensors(accelerometer, gyroscope, etc.). – 0x00000800 - to receive time sync requests from the GPS engine. Time sync enables the GPS engine to synchronize its clock with the sensor processor's clock. – 0x00001000 - to receive Stationary Position Indicator (SPI) streaming report indications. – 0x00002000 - to receive location server requests. These requests are generated when the service wishes to establish a connection with a location server. – 0x00004000 - to receive notifications related to network-initiated Geofences. These events notify the client when a network-initiated Geofence is added, deleted, or edited. – 0x00008000 - to receive Geofence alerts. These alerts are generated to inform the client of the changes that may affect a Geofence, e.g., if GPS is turned off or if the network is unavailable. – 0x00010000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach report is for a single Geofence. – 0x00020000 - to register for pedometer control requests from the location engine. The location engine sends this event to control the injection of pedometer reports. – 0x00040000 - to register for motion data control requests from the location engine. The location engine sends this event to control the injection of motion data. – 0x00080000 - to receive notification when a batch is full. The location engine sends this event to notify of Batch Full for ongoing batching session. – 0x00100000 - to receive position report indications along with an ongoing batching session. The location engine sends this event to notify the batched position report while a batching session is ongoing. – 0x00200000 - to receive Wi-Fi Access Point (AP) data inject request event indications. – 0x00400000 - to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach notification is for multiple Geofences. Breaches from multiple Geofences are all batched and sent in the same notification. – 0x00800000 - to receive notifications from the location engine indicating its readiness to accept vehicle data (vehicle accelerometer, vehicle angular rate, vehicle odometry, etc.). – 0x01000000 - to receive system clock and satellite measurement report events (system clock, SV time, Doppler, etc.). – 0x02000000 - to receive satellite position reports as polynomials.Reports are generated only for the GNSS satellite constellations that are enabled using QMI_LOC_SET_GNSS_-CONSTELL_REPORT_CONFIG.

Note

Multiple events can be registered by OR the individual masks and sending them in this TLV. All unused bits in this mask must be set to 0.

Parameters

TlvResult	• Pack result.

8.252.2 Field Documentation

8.252.2.1 uint64_t pack_loc_EventRegister_t::eventRegister

8.252.2.2 uint16_t pack_loc_EventRegister_t::TlvResult

8.253 pack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint32_t extPowerState
- uint16_t TlvResult

8.253.1 Detailed Description

This structure contains the Parameter External Power Source State pack.

Parameters

extPowerState	• Specifies the Power state; injected by the control point. • Values – 0 - Device is not connected to an external power source – 1 - Device is connected to an external power source – 2 - Unknown external power state
TlvResult	• Pack result.

8.253.2 Field Documentation

8.253.2.1 uint32_t pack_loc_SetExtPowerState_t::extPowerState

8.253.2.2 uint16_t pack_loc_SetExtPowerState_t::TlvResult

8.254 pack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t TlvResult

8.254.1 Detailed Description

This structure contains Set Operation Mode pack

Parameters

mode	• Valid values: – eQMI_LOC_OPER_MODE_DEFAULT (1) - Use the default engine mode – eQMI_LOC_OPER_MODE_MSB (2) - Use the MS-based mode – eQMI_LOC_OPER_MODE_MSA (3) - Use the MS-assisted mode – eQMI_LOC_OPER_MODE_STANDALONE (4) - Use Standalone mode – eQMI_LOC_OPER_MODE_CELL_ID (5) - Use cell ID; this mode is only valid for GSM/UMTS networks – eQMI_LOC_OPER_MODE_WWAN (6) - Use WWAN measurements to calculate the position; if this mode is set, AFLT will be used for 1X networks and OTDOA will be used for LTE networks
Tlvresult	• Pack result.

8.254.2 Field Documentation

8.254.2.1 uint32_t pack_loc_SetOperationMode_t::mode

8.254.2.2 uint16_t pack_loc_SetOperationMode_t::Tlvresult

8.255 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint32_t xid
- uint16_t Tlvresult

8.255.1 Detailed Description

This structure contains Set Operation Mode pack

Parameters

xid	• Identifies the transaction. • The transaction ID is returned in the Get Best Available Position indication.
Tlvresult	• Pack result.

8.255.2 Field Documentation

8.255.2.1 uint16_t pack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.255.2.2 uint32_t pack_loc_SLQSLOCGetBestAvailPos_t::xid

8.256 pack_loc_SLQSLOCGetServer_t Struct Reference

Data Fields

- uint32_t serverType
- int has_serverAddrTypeMask
- uint8_t serverAddrTypeMask

8.256.1 Detailed Description

This structure contains parameters to Gets the location server. Please check has_<Param_Name> field for presence of optional parameters

Parameters

serverType	• Type of server. • Valid values – eQMI_LOC_SERVER_TYPE_CDMA_PDE (1) - Server type is CDMA PDE – eQMI_LOC_SERVER_TYPE_CDMA_MPC (2) - Server type is CDMA MPC – eQMI_LOC_SERVER_TYPE_UMTS_SLP (3) - Server type is UMTS SLP – eQMI_LOC_SERVER_TYPE_CUSTOM_PDE (4)- Server type is custom PDE
serverAddrTypeMask	• Optional parameter • Type of address the client wants. If unspecified, the indication will contain all the types of addresses it has for the specified server type. • Valid bitmasks – 0x01 – IPv4 – 0x02 – IPv6 – 0x04 – URL

8.256.2 Field Documentation

8.256.2.1 int pack_loc_SLQSLOCGetServer_t::has_serverAddrTypeMask

8.256.2.2 uint8_t pack_loc_SLQSLOCGetServer_t::serverAddrTypeMask

8.256.2.3 uint32_t pack_loc_SLQSLOCGetServer_t::serverType

8.257 pack_loc_SLQSLOCInjectPosition_t Struct Reference

Data Fields

- double latitude
- int has_latitude
- double longitude
- int has_longitude
- float horUncCircular

- int has_horUncCircular
- uint8_t horConfidence
- int has_horConfidence
- uint32_t horReliability
- int has_horReliability
- float altitudeWrtEllipsoid
- int has_altitudeWrtEllipsoid
- float altitudeWrtMeanSeaLevel
- int has_altitudeWrtMeanSeaLevel
- float vertUnc
- int has_vertUnc
- uint8_t vertConfidence
- int has_vertConfidence
- uint32_t vertReliability
- int has_vertRelicability
- altSrcInfo_t altitudeSrcInfo
- int has_altitudeSrcInfo
- uint64_t timestampUtc
- int has_timestampUtc
- uint32_t timestampAge
- int has_timestampAge
- uint32_t positionSrc
- int has_positionSrc
- float rawHorUncCircular
- int has_rawHorUncCircular
- uint8_t rawHorConfidence
- int has_rawHorConfidence

8.257.1 Detailed Description

This structure contains LOC Inject Position parameters Please check has_<Param_Name> field for presence of optional parameters

Parameters

latitude	• Optional parameter • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude • Note - This field must be specified together with pLongitude and pHorUncCircular.
longitude	• Optional parameter • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude • Note - This field must be specified together with pLatitude and pHorUncCircular.

horUncCircular	• Optional parameter • Horizontal position uncertainty. • Units - Meters • Note - This field must be specified together with pLatitude and pLongitude.
horConfidence	• Optional parameter • Horizontal confidence. • Units - Percent • Values – Valid Values - 1 to 99 – Invalid Values - 0, 101 to 255 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with horizontal uncertainty. If not specified when pHorUncCircular is set, the default value is 50.
horReliability	• Optional parameter – Values * 0 - Location reliability is not set. * 1 - Location reliability is very low; use it at your own risk * 2 - Location reliability is low; little or no cross-checking is possible. * 3 - Location reliability is medium; limited cross-check passed * 4 - Location reliability is high; strong cross-check passed
altitudeWrt-Ellipsoid	• Optional parameter • Altitude With Respect to Ellipsoid. • Units - Meters • Values – Positive - height – Negative = depth
altitudeWrt-MeanSeaLevel	• Optional parameter • Altitude With Respect to Sea Level. • Units - Meters
vertUnc	• Optional parameter • Vertical uncertainty. • Units - Meters • Note - This is mandatory if either pAltitudeWrtEllipsoid or pAltitudeWrtMeanSeaLevel is specified.

vertConfidence	• Optional parameter • Vertical confidence. • Units - Percentage • Values – Valid Values - 0 to 99 – Invalid Values - 0, 100-256 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with the vertical uncertainty. If not specified, the default value will be 50.
vertReliability	• Optional parameter – Values * 0 - Location reliability is not set. * 1 - Location reliability is very low; use it at your own risk. * 2 - Location reliability is low; little or no cross-checking is possible * 3 - Location reliability is medium; limited cross-check passed * 4 - Location reliability is high; strong cross-check passed
altitudeSrcInfo	• Optional parameter – Pointer to struct altitudeSrcInfo. See altitudeSrcInfo for more information
timestampUtc	• Optional parameter • UTC timestamp • Units - Milliseconds since Jan. 1, 1970
timestampAge	• Optional parameter • Position age, which is an estimate of how long ago this fix was made. • Units - Milliseconds
positionSrc	• Optional parameter • Source from which this position was obtained • Valid values – 0 - Position source is GNSS – 1 - Position source is Cell ID – 2 - Position source is Enhanced Cell ID – 3 - Position source is Wi-Fi – 4 - Position source is Terrestrial – 5 - Position source is GNSS Terrestrial Hybrid – 6 - Other sources • Note - If altitude is specified and the altitude source is not specified, the engine assumes that the altitude was obtained using the specified position source. – If both altitude and altitude source are specified, the engine assumes that only latitude and longitude were obtained using the specified position source.

rawHorUncCircular	• Optional parameter • Horizontal position uncertainty (circular) without any optimization. • Units - Meters
rawHorConfidence	• Optional parameter • Horizontal confidence associated with raw horizontal uncertainty • Units: Percent • Values – Valid values - 1 to 99 – Invalid values - 0, 101 to 255 – If 100 is received, reinterpret to 99 • Note - This field must be specified together with raw horizontal uncertainty. If not specified when rawHorUncCircular is set, the default value is 50.

8.257.2 Field Documentation

8.257.2.1 altSrcInfo_t pack_loc_SLQSLOCInjectPosition_t::altitudeSrcInfo

8.257.2.2 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtEllipsoid

8.257.2.3 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtMeanSeaLevel

8.257.2.4 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeSrcInfo

8.257.2.5 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtEllipsoid

8.257.2.6 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtMeanSeaLevel

8.257.2.7 int pack_loc_SLQSLOCInjectPosition_t::has_horConfidence

8.257.2.8 int pack_loc_SLQSLOCInjectPosition_t::has_horReliability

8.257.2.9 int pack_loc_SLQSLOCInjectPosition_t::has_horUncCircular

8.257.2.10 int pack_loc_SLQSLOCInjectPosition_t::has_latitude

8.257.2.11 int pack_loc_SLQSLOCInjectPosition_t::has_longitude

8.257.2.12 int pack_loc_SLQSLOCInjectPosition_t::has_positionSrc

8.257.2.13 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorConfidence

8.257.2.14 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorUncCircular

8.257.2.15 int pack_loc_SLQSLOCInjectPosition_t::has_timestampAge

8.257.2.16 int pack_loc_SLQSLOCInjectPosition_t::has_timestampUtc

8.257.2.17 int pack_loc_SLQSLOCInjectPosition_t::has_vertConfidence

8.257.2.18 int pack_loc_SLQSLOCInjectPosition_t::has_vertRelicability

8.257.2.19 int pack_loc_SLQSLOCInjectPosition_t::has_vertUnc

8.257.2.20 uint8_t pack_loc_SLQSLOCInjectPosition_t::horConfidence

8.257.2.21 uint32_t pack_loc_SLQSLOCInjectPosition_t::horReliability

8.257.2.22 float pack_loc_SLQSLOCInjectPosition_t::horUncCircular

8.257.2.23 double pack_loc_SLQSLOCInjectPosition_t::latitude

8.257.2.24 double pack_loc_SLQSLOCInjectPosition_t::longitude

8.257.2.25 uint32_t pack_loc_SLQSLOCInjectPosition_t::positionSrc

8.257.2.26 uint8_t pack_loc_SLQSLOCInjectPosition_t::rawHorConfidence

8.257.2.27 float pack_loc_SLQSLOCInjectPosition_t::rawHorUncCircular

8.257.2.28 uint32_t pack_loc_SLQSLOCInjectPosition_t::timestampAge

8.257.2.29 uint64_t pack_loc_SLQSLOCInjectPosition_t::timestampUtc

8.257.2.30 uint8_t pack_loc_SLQSLOCInjectPosition_t::vertConfidence

8.257.2.31 uint32_t pack_loc_SLQSLOCInjectPosition_t::vertReliability

8.257.2.32 float pack_loc_SLQSLOCInjectPosition_t::vertUnc

8.258 pack_loc_SLQSLOCInjectSensorData_t Struct Reference

Data Fields

- int has_opaqueId
- uint32_t opaqueId
- int has_accleroData
- sensorData_t acceleroData
- int has_gyroData
- sensorData_t gyroData
- int has_acceleroTimeSrc
- uint32_t acceleroTimeSrc
- int has_gyroTimeSrc
- uint32_t gyroTimeSrc
- int has_accelTemp
- tempData_t accelTemp
- int has_gyroTemp
- tempData_t gyroTemp

8.258.1 Detailed Description

This structure contains parameters to inject sensor data into the GNSS location engine Please check has_<Param-_Name> field for presence of optional parameters

Parameters

opaqueId	• Opaque Identifier (Optional parameter) • An opaque identifier that is sent in by the client that will be echoed in the indication so the client can relate the indication to the request.
acceleroData	• 3-Axis Accelerometer Data (Optional parameter) • Pointer to struct sensorData. See sensorData_t for more information
gyroData	• 3-Axis Cyroscope Data (Optional parameter) • Pointer to struct sensorData. See sensorData_t for more information
acceleroTimeSrc	• 3-Axis Accelerometer Data Time Source (Optional parameter) • The location service uses this field to identify the time reference used in the accelerometer data time stamps. • If not specified, the location service assumes that the time source for the accelerometer data is unknown. • Valid values – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
gyroTimeSrc	• 3-Axis Gyroscope Data Time Source (Optional) • The location service uses this field to identify the time reference used in the gyroscope data time stamps. • If not specified, the location service assumes that the time source for the gyroscope data is unknown. • Valid values – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine
accelTemp	• Accelerometer Temperature Data (Optional parameter) • Pointer to struct tempratureData. See tempData_t for more information
gyroTemp	• Gyroscope Temperature Data (Optional parameter) • Pointer to struct tempratureData. See tempData_t for more information

8.258.2 Field Documentation

8.258.2.1 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::acceleroData

8.258.2.2 uint32_t pack_loc_SLQSLOCInjectSensorData_t::acceleroTimeSrc

8.258.2.3 tempData_t pack_loc_SLQSLOCInjectSensorData_t::accelTemp

8.258.2.4 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::gyroData

8.258.2.5 tempData_t pack_loc_SLQSLOCInjectSensorData_t::gyroTemp

8.258.2.6 uint32_t pack_loc_SLQSLOCInjectSensorData_t::gyroTimeSrc

8.258.2.7 int pack_loc_SLQSLOCInjectSensorData_t::has_acceleroTimeSrc

8.258.2.8 int pack_loc_SLQSLOCInjectSensorData_t::has_accelTemp

8.258.2.9 int pack_loc_SLQSLOCInjectSensorData_t::has_accleroData

8.258.2.10 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroData

8.258.2.11 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTemp

8.258.2.12 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTimeSrc

8.258.2.13 int pack_loc_SLQSLOCInjectSensorData_t::has_opaqueId

8.258.2.14 uint32_t pack_loc_SLQSLOCInjectSensorData_t::opaqueId

8.259 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference

Data Fields

- uint64_t timeMsec
- uint32_t timeUncMsec

8.259.1 Detailed Description

Parameters

timeMsec	• The UTC time since Jan. 1, 1970
timeUncMsec	• The time Uncertainty

8.259.2 Field Documentation

8.259.2.1 uint64_t pack_loc_SLQSLOCInjectUTCTime_t::timeMsec

8.259.2.2 uint32_t pack_loc_SLQSLOCInjectUTCTime_t::timeUncMsec

8.260 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference

Data Fields

- uint32_t state
- int has_confidence
- uint8_t confidence

8.260.1 Detailed Description

This structure contains parameters to set current cradle mount configuration Please check has_<Param_Name> field for presence of optional parameters

Parameters

state	• Cradle Mount State • Valid values: – 0 - Device is mounted on the cradle – 1 - Device is not mounted on the cradle – 2 - Unknown cradle mount state
confidence	• Cradle Mount Confidence (Optional) • Confidence in the Cradle Mount state expressed as a percentage. • Range - 0 to 100

8.260.2 Field Documentation

8.260.2.1 uint8_t pack_loc_SLQSLOCSetCradleMountConfig_t::confidence

8.260.2.2 int pack_loc_SLQSLOCSetCradleMountConfig_t::has_confidence

8.260.2.3 uint32_t pack_loc_SLQSLOCSetCradleMountConfig_t::state

8.261 pack_loc_Start_t Struct Reference

Data Fields

- uint8_t SessionId
- uint32_t * pRecurrenceType
- uint32_t * pHorizontalAccuracyLvl
- uint32_t * pIntermediateReportState
- uint32_t * pMinIntervalTime
- loc_LocApplicationInfo * pApplicationInfo
- uint32_t * pConfigAltitudeAssumed
- uint16_t Tlvresult

8.261.1 Detailed Description

This structure contains the LOC Start pack

Parameters

SessionId	• ID of the session as identified by the control point. • Range: 0 to 255

pRecurrence-Type	• Optional Parameter • Specifies the type of session in which the control point is interested. • Defaults to SINGLE. -Values – 1 - Request periodic position fixes – 2 - Request a single position fix
pHorizontal-AccuracyLvl	• Optional Parameter • Specifies the horizontal accuracy level required by the control point. • Defaults to LOW • Values – 1 - Low accuracy – 2 - Medium accuracy – 3 - High accuracy
pIntermediate-ReportState	• Optional Parameter • Specifies if the control point is interested in receiving intermediate reports. • ON by default. • Values – 1 - Intermediate reports are turned on – 2 - Intermediate reports are turned off
pMinInterval-Time	• Optional Parameter • Minimum time interval, specified by the control point, that must elapse between position reports. • Units - Milliseconds • Default - 1000 ms
LocApplication-Info	• Optional Parameter • LOC Application Parameters • See loc_LocApplicationInfo for more information
pConfigAltitude-Assumed	• Optional Parameter • Configuration for Altitude Assumed Info in GNSS SV Info Event • Defaults to ENABLED. • Values – 1 - Enable Altitude Assumed information in GNSS SV Info Event – 2 - Disable Altitude Assumed information in GNSS SV Info Event
Tlvresult	• Unpack result.

8.261.2 Field Documentation

8.261.2.1 loc_LocApplicationInfo* pack_loc_Start_t::pApplicationInfo

8.261.2.2 uint32_t* pack_loc_Start_t::pConfigAltitudeAssumed

8.261.2.3 uint32_t* pack_loc_Start_t::pHorizontalAccuracyLvl

8.261.2.4 uint32_t* pack_loc_Start_t::pIntermediateReportState

8.261.2.5 uint32_t* pack_loc_Start_t::pMinIntervalTime

8.261.2.6 uint32_t* pack_loc_Start_t::pRecurrenceType

8.261.2.7 uint8_t pack_loc_Start_t::SessionId

8.261.2.8 uint16_t pack_loc_Start_t::Tlvresult

8.262 pack_loc_Stop_t Struct Reference

Data Fields

- uint8_t SessionId
- uint16_t Tlvresult

8.262.1 Detailed Description

This structure contains Stop LOC pack

Parameters

sessionId	• ID of the session as identified by the control point. • Range: 0 to 255
Tlvresult	• Unpack result.

8.262.2 Field Documentation

8.262.2.1 uint8_t pack_loc_Stop_t::SessionId

8.262.2.2 uint16_t pack_loc_Stop_t::Tlvresult

8.263 pack_nas_InitiateDomainAttach_t Struct Reference

Data Fields

- uint32_t action

8.263.1 Detailed Description

Pack structure for Initiates a domain attach/detach of the device.

Parameters

action	• Domain action to attempt 1 - Attach 2 - Detach

8.263.2 Field Documentation

8.263.2.1 uint32_t pack_nas_InitiateDomainAttach_t::action

8.264 pack_nas_SetACCOLC_t Struct Reference

Data Fields

- int8_t spc [6]
- uint8_t accolc

8.264.1 Detailed Description

Sets the access overload class (ACCOLC)

Parameters

spc	• service programming code NULL-terminated string of six digit
accolc	• ACCOLC : Valid range is 0 to 15

8.264.2 Field Documentation

8.264.2.1 uint8_t pack_nas_SetACCOLC_t::accolc

8.264.2.2 int8_t pack_nas_SetACCOLC_t::spc[6]

8.265 pack_nas_SetCDMANetworkParameters_t Struct Reference

Data Fields

- char * pSPC
- uint8_t * pForceRev0
- uint8_t * pCustomSCP
- uint32_t * pProtocol
- uint32_t * pBroadcast
- uint32_t * pApplication
- uint32_t * pRoaming

8.265.1 Detailed Description

Pack structure to set CDMA Network parameters.

Parameters

pSPC	• Six digit service programming code (not necessary when only the roaming field is being set)
pForceRev0	• (Optional)Force CDMA 1x-EV-DO Rev. 0 mode – 0 - Disabled – 1 - Enabled Note: Enabled can only be specified if pCustomSCP state is set to Disabled
pCustomSCP	• (Optional)Use a custom config for CDMA 1x-EV-DO SCP – 0 - Disabled – 1 - Enabled Note: Enabled can only be specified if pForceRev0 is set to Disabled
pProtocol	• Protocol mask for custom SCP config – 0x00000001 - Subtype 2 Physical Layer – 0x00000002 - Enhanced CCMAC – 0x00000004 - Enhanced ACMAC – 0x00000008 - Enhanced FTCMAC – 0x00000010 - Subtype 3 RTCMAC – 0x00000020 - Subsystem 1 RTCMAC – 0x00000040 - Enhanced Idle – 0x00000080 - Generic Multimode Capable Disc Port – 0xFFFFFFFF - Unknown
pBroadcast	• Broadcast mask for custom SCP config – 0x00000001 - Generic broadcast enabled – 0xFFFFFFFF - Unknown
pApplication	• Application mask for custom SCP config – 0x00000001 - SN Multiflow Packet Application – 0x00000002 - SN Enhanced Multiflow Packet Application – 0xFFFFFFFF - Unknown
pRoaming	• Roaming preference – 0 - Automatic – 1 - Home Only – 2 - Affiliated Roaming Only – 3 - Home and Affiliated Roaming – 0xFFFFFFFF - Unknown

8.265.2 Field Documentation

8.265.2.1 uint32_t* pack_nas_SetCDMANetworkParameters_t::pApplication

8.265.2.2 uint32_t* pack_nas_SetCDMANetworkParameters_t::pBroadcast

8.265.2.3 uint8_t* pack_nas_SetCDMANetworkParameters_t::pCustomSCP

8.265.2.4 uint8_t* pack_nas_SetCDMANetworkParameters_t::pForceRev0

8.265.2.5 uint32_t* pack_nas_SetCDMANetworkParameters_t::pProtocol

8.265.2.6 uint32_t* pack_nas_SetCDMANetworkParameters_t::pRoaming

8.265.2.7 char* pack_nas_SetCDMANetworkParameters_t::pSPC

8.266 pack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint32_t TechnologyPref
- uint32_t Duration
- uint16_t Tlvresult

8.266.1 Detailed Description

Parameters

TechnologyPref[-IN]	• Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved
Duration[IN]	• Duration of active preference – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call
Tlvresult	• pack result

8.266.2 Field Documentation

8.266.2.1 uint32_t pack_nas_SetNetworkPreference_t::Duration

8.266.2.2 uint32_t pack_nas_SetNetworkPreference_t::TechnologyPref

8.266.2.3 uint16_t pack_nas_SetNetworkPreference_t::TlvResult

8.267 pack_nas_SLQSConfigSigInfo_t Struct Reference

Data Fields

- nas_RSSIThresh * pRSSIThresh
- nas_ECIOThresh * pECIOThresh
- nas_HDRSINRThresh * pHDRSINRThresh
- nas_LTESNRThresh * pLTESNRThresh
- nas_IOThresh * pIOThresh
- nas_RSRQThresh * pRSRQThresh
- nas_RSRPThresh * pRSRPThresh
- nas_LTESigRptCfg * pLTESigRptCfg
- nas_TDSCDMASINRCONFThresh * pTDSCDMASINRCONFThresh

8.267.1 Detailed Description

Structure for ConfigSigInfo pack.

Parameters

pRSSIThresh	• RSSI threshold List • See nas_RSSIThresh for more details
pECIOThresh	• ECIO Threshold List • See nas_ECIOThresh for more details
pHDRSINR-Thresh	• HDR SINR Threshold List • See nas_HDRSINRThresh for more details
pLTESNR-Thresh	• LTE SNR Threshold List • See nas_LTESNRThresh for more details
pIOThresh	• IO Threshold List • See nas_IOThresh for more details
pRSRQThresh	• RSRQ Threshold List • See nas_RSRQThresh for more details
pRSRPThresh	• RSRP Threshold List • See nas_RSRPThresh for more details
pLTESigRptCfg	• LTE signal report config • See nas_LTESigRptCfg for more details

pTDSCDMASINRCONFThresh	• TD-SCDMA SINR Threshold List • See nas_TDSCDMASINRCONFThresh for more details

8.267.2 Field Documentation

8.267.2.1 nas_ECIOThresh∗ pack_nas_SLQSConfigSigInfo_t::pECIOThresh

8.267.2.2 nas_HDRSINRThresh∗ pack_nas_SLQSConfigSigInfo_t::pHDRSINRThresh

8.267.2.3 nas_IOThresh∗ pack_nas_SLQSConfigSigInfo_t::pIOThresh

8.267.2.4 nas_LTESigRptCfg∗ pack_nas_SLQSConfigSigInfo_t::pLTESigRptCfg

8.267.2.5 nas_LTESNRThresh∗ pack_nas_SLQSConfigSigInfo_t::pLTESNRThresh

8.267.2.6 nas_RSRPThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRPThresh

8.267.2.7 nas_RSRQThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRQThresh

8.267.2.8 nas_RSSIThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSSIThresh

8.267.2.9 nas_TDSCDMASINRCONFThresh∗ pack_nas_SLQSConfigSigInfo_t::pTDSCDMASINRCONFThresh

8.268 pack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t ∗ pMncPcsStatus

8.268.1 Detailed Description

Parameters

mcc	• A 16-bit integer representation of MCC. Range: 0 to 999
mnc	• A 16-bit integer representation of MNC. Range: 0 to 999
pMncPcsStatus	• MNC PCS Digit Include Status • Used to interpret the length of the corresponding MNC reported in the PLMN TLV(0x01). • Values – TRUE - MNC is a three-digit value. e.g. a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a two-digit value. e.g. a reported value of 90 corresponds to an MNC value of 90

Note

If pMncPcsStatus is not present, an MNC smaller than 100 is assumed to be a two-digit value, and an MNC greater than or equal to 100 is assumed to be a three digit value.

8.268.2 Field Documentation

8.268.2.1 uint16_t pack_nas_SLQSGetPLMNName_t::mcc

8.268.2.2 uint16_t pack_nas_SLQSGetPLMNName_t::mnc

8.268.2.3 uint8_t* pack_nas_SLQSGetPLMNName_t::pMncPcsStatus

8.269 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference

Data Fields

- uint32_t regAction
- nas_MNRInfo * pMNRInfo
- uint32_t * pChangeDuration
- uint8_t * pMncPcsDigitStatus

8.269.1 Detailed Description

This structure contains Initiate Network Registration request parameters

Parameters

regAction	• Specifies one of the following register actions : – AUTO_REGISTER - Device registers according to its provisioning and optional parameters supplied with the command are ignored. – MANUAL_REGISTER - Device registers to a specified network and the optional Manual Network Register Information parameter pMNRInfo must also be included for the command to process successfully and supported only for 3GPP.
pMNRInfo	[Optional] • Pointer to structure MNRInfo – See nas_MNRInfo for more information
pChange-Duration	[Optional] • Duration of the change. – 0x00 - Power cycle - Remains active until the next device power cycle – 0x01 - Permanent - Remains active through power cycles until changed by the client
pMncPcsDigit-Status	[Optional] • MNC PCS Digit Include Status – True - MNC is a 3-digit value. – False - MNC is a 2-digit value.

8.269.2 Field Documentation

8.269.2.1 uint32_t* pack_nas_SLQSInitiateNetworkRegistration_t::pChangeDuration

8.269.2.2 uint8_t* pack_nas_SLQSInitiateNetworkRegistration_t::pMncPcsDigitStatus

8.269.2.3 nas_MNRInfo* pack_nas_SLQSInitiateNetworkRegistration_t::pMNRInfo

8.269.2.4 uint32_t pack_nas_SLQSInitiateNetworkRegistration_t::regAction

8.270 pack_nas_SLQSNasConfigSigInfo2_t Struct Reference

Data Fields

- nas_CDMARSSIThresh * pCDMARSSIThresh
- uint16_t * pCDMARSSIDelta
- nas_CDMAECIOThresh * pCDMAECIOThresh
- uint16_t * pCDMAECIODelta
- nas_HDRRSSIThresh * pHDRRSSIThresh
- uint16_t * pHDRRSSIDelta
- nas_HDRECIOThresh * pHDRECIOThresh
- uint16_t * pHDRECIODelta
- nas_HDRSINRThreshold * pHDRSINRThresh
- uint16_t * pHDRSINRDelta
- nas_HDRIOThresh * pHDRIOThresh
- uint16_t * pHDRIODelta
- nas_GSMRSSIThresh * pGSMRSSIThresh
- uint16_t * pGSMRSSIDelta
- nas_WCDMARSSIThresh * pWCDMARSSIThresh
- uint16_t * pWCDMARSSIDelta
- nas_WCDMAECIOThresh * pWCDMAECIOThresh
- uint16_t * pWCDMAECIODelta
- nas_LTERSSIThresh * pLTERSSIThresh
- uint16_t * pLTERSSIDelta
- nas_LTESNRThreshold * pLTESNRThresh
- uint16_t * pLTESNRDelta
- nas_LTERSRQThresh * pLTERSRQThresh
- uint16_t * pLTERSRQDelta
- nas_LTERSRPThresh * pLTERSRPThresh
- uint16_t * pLTERSRPDelta
- nas_LTESigRptConfig * pLTESigRptConfig
- nas_TDSCDMARSCPThresh * pTDSCDMARSCPThresh
- uint16_t * pTDSCDMARSCPDelta
- nas_TDSCDMARSSIThresh * pTDSCDMARSSIThresh
- float * pTDSCDMARSSIDelta
- nas_TDSCDMAECIOThresh * pTDSCDMAECIOThresh
- float * pTDSCDMAECIODelta
- nas_TDSCDMASINRThresh * pTDSCDMASINRThresh
- float * pTDSCDMASINRDelta

8.270.1 Detailed Description

Parameters

pCDMARSSI-Thresh	• CDMA RSSI threshold List
pCDMARSSI-Delta	• RSSI delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pCDMAECIO-Thresh	• CDMA ECIO Threshold List
pCDMAECIO-Delta	• ECIO delta (in units of 0.1 dB). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRRSSI-Thresh	• HDR RSSI Threshold List
pHDRRSSIDelta	• RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRECIO-Thresh	• HDR ECIO Threshold List
pHDRECIODelta	• ECIO delta (in units of 0.1 dB) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRSINR-Thresh	• HDR SINR Threshold List
pHDRSINRDelta	• SINR delta (in units of 1 SINR level) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pHDRIOThresh	• HDR IO Threshold List
pHDRIODelta	• IO delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pGSMRSSI-Thresh	• GSM RSSI Threshold List • See nas_GSMRSSIThresh for more details
pGSMRSSIDelta	• RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.

pWCDMARSSI-Thresh	• WCDMA RSSI Threshold List • See nas_WCDMARSSIThresh for more details
pWCDMARSSI-Delta	• RSSI delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pWCDMAECIO-Thresh	• WCDMA ECIO Threshold List
pWCDMAECIO-Delta	• ECIO delta (in units of 0.1 dB) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTERSSI-Thresh	• LTE RSSI Threshold List
pLTERSSIDelta	• RSSI delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTESNR-Thresh	• LTE SNR Threshold List
pLTESNRDelta	• SNR delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTERSRQ-Thresh	• LTE RSRQ Threshold List
pLTERSRQ-Delta	• RSRQ delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTERSRP-Thresh	• LTE RSRP Threshold List
pLTERSRPDelta	• RSRP delta (in units of 0.1 dBm). • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.
pLTESigRpt-Config	• LTE Signal Report Config
pTDSCDMARS-CPThresh	• TDSCDMA RSCP Threshold List
pTDSCDMARS-CPDelta	• RSCP delta (in units of 0.1 dBm) • A value of 0 is rejected with a QMI_ERR_INVALID_ARG error.

pTDSCDMARSSIThresh	• TDSCDMA RSSI Threshold List
pTDSCDMARSSIDelta	• RSSI delta (in dBm) used by TD-SCDMA.
pTDSCDMAECIOThresh	• TDSCDMA ECIO Threshold List
pTDSCDMAECIODelta	• ECIO delta (in dB) used by TD-SCDMA
pTDSCDMASINRThresh	• TDSCDMA SINR Threshold List
pTDSCDMASINRDelta	• SINR delta (in dB) used by TD-SCDMA.

Note

Mixture of threshold and delta values can be provided in the request. But for each type and RAT,only one of threshold list or delta value is to be provided.

8.270.2 Field Documentation

8.270.2.1 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIODelta

8.270.2.2 nas_CDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIOThresh

8.270.2.3 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIDelta

8.270.2.4 nas_CDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIThresh

8.270.2.5 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIDelta

8.270.2.6 nas_GSMRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIThresh

8.270.2.7 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIODelta

8.270.2.8 nas_HDRECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIOThresh

8.270.2.9 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIODelta

8.270.2.10 nas_HDRIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIOThresh

8.270.2.11 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIDelta

8.270.2.12 nas_HDRRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIThresh

8.270.2.13 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRDelta

8.270.2.14 nas_HDRSINRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRThresh

8.270.2.15 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPDelta

8.270.2.16 nas_LTERSRPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPThresh

8.270.2.17 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQDelta

8.270.2.18 nas_LTERSRQThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQThresh

8.270.2.19 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIDelta

8.270.2.20 nas_LTERSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIThresh

8.270.2.21 nas_LTESigRptConfig* pack_nas_SLQSNasConfigSigInfo2_t::pLTESigRptConfig

8.270.2.22 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRDelta

8.270.2.23 nas_LTESNRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRThresh

8.270.2.24 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIODelta

8.270.2.25 nas_TDSCDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIOThresh

8.270.2.26 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPDelta

8.270.2.27 nas_TDSCDMARSCPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPThresh

8.270.2.28 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIDelta

8.270.2.29 nas_TDSCDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIThresh

8.270.2.30 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRDelta

8.270.2.31 nas_TDSCDMASINRThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRThresh

8.270.2.32 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIODelta

8.270.2.33 nas_WCDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIOThresh

8.270.2.34 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIDelta

8.270.2.35 nas_WCDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIThresh

8.271 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference

Data Fields

- uint8_t namID

8.271.1 Detailed Description

Structure for 3GPP2 Subscription pack.

Parameters

namID	[Mandatory] • NAM ID of the information to be retrieved. The index starts from 0. A nam_id of 0xFF is used to retrieve information of current NAM.

8.271.2 Field Documentation

8.271.2.1 uint8_t pack_nas_SLQSNasGet3GPP2Subscription_t::namID

8.272 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- uint8_t radio_if

8.272.1 Detailed Description

Structure for TxRx Info pack

Parameters

radio_if	[Mandatory] • Radio interface technology of the signal being measured • Valid Values – 0x01 - NAS_RADIO_IF_CDMA_1X - CDMA – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - HDR – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE

8.272.2 Field Documentation

8.272.2.1 uint8_t pack_nas_SLQSNasGetTxRxInfo_t::radio_if

8.273 pack_nas_SLQSNasIndicationRegisterExt_t Struct Reference

Data Fields

- uint8_t * pSystemSelectionInd
- uint8_t * pDDTMInd
- uint8_t * pServingSystemInd
- uint8_t * pDualStandByPrefInd
- uint8_t * pSubscriptionInfoInd
- uint8_t * pNetworkTimeInd
- uint8_t * pSysInfoInd
- uint8_t * pSignalStrengthInd
- uint8_t * pErrorRateInd
- uint8_t * pHDRNewUATIAssInd
- uint8_t * pHDRSessionCloseInd
- uint8_t * pManagedRoamingInd

- uint8_t * pNetworkRejectInd
- uint8_t * pSuppressSysInfoInd
- uint8_t * pLTECphyCa

8.273.1 Detailed Description

Parameters

pSystemSelectionInd	[Optional] • System Selection Preference indication registration. The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pDDTMInd	[Optional] • DDTM (Data Dedicated Transmission Mode) indication registration. The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pServingSystemInd	[Optional] • Serving System indication registration. The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pDualStandByPrefInd	[Optional] • Dual Standby Preference indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSubscriptionInfoInd	[Optional] • Subscription Information indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pNetworkTimeInd	[Optional] • Network Time indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSysInfoInd	[Optional] • System Information indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable

pSignalStrength-Ind	[Optional] • Signal Strength indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pErrorRateInd	[Optional] • Error Rate indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pHDRNewUATI-AssInd	[Optional] • HDR New UATI Assigned indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pHDRSession-CloseInd	[Optional] • HDR Session Closed indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pManaged-RoamingInd	[Optional] • Managed Roaming indication registration.The callbacks would not be invoked if the indication is disabled. – 0x00 - Disable
pNetworkReject-Ind	[Optional] • Network reject information indication registration.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable – 0x01 - Enable
pSuppressSys-InfoInd	[Optional] • Controls the reporting of QMI_NAS_SYS_INFO_IND when only the reject_cause field has changed. Values: – 0x00 - Disable – 0x01 - Enable- 0x01 - Enable
pLTECphyCa	[Optional] • LTE Physical Carrier Aggregation Information.The following callbacks would not be invoked if the indication is disabled. – 0x00 - Disable (default value) – 0x01 - Enable

8.273.2 Field Documentation

8.273.2.1 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDDTMInd

8.273.2.2 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDualStandByPrefInd

8.273.2.3 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pErrorRateInd

8.273.2.4 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRNewUATIAssInd

8.273.2.5 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRSessionCloseInd

8.273.2.6 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pLTECphyCa

8.273.2.7 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pManagedRoamingInd

8.273.2.8 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkRejectInd

8.273.2.9 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkTimeInd

8.273.2.10 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pServingSystemInd

8.273.2.11 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSignalStrengthInd

8.273.2.12 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSubscriptionInfoInd

8.273.2.13 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSuppressSysInfoInd

8.273.2.14 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSysInfoInd

8.273.2.15 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSystemSelectionInd

8.274 pack_nas_SLQSNasSwiIndicationRegister_t Struct Reference

Data Fields

- uint8_t lteEsmUl
- uint8_t lteEsmDl
- uint8_t lteEmmUl
- uint8_t lteEmmDl
- uint8_t gsmUmtsUl
- uint8_t gsmUmtsDl
- uint8_t * pRankIndicatorInd
- uint8_t * pTimer

8.274.1 Detailed Description

This structure contains the OTA message indication.

Parameters

lteEsmUl	• 0 - do not report • 1 - report LTE ESM uplink messages
lteEsmDl	• 0 - do not report • 1 - report LTE ESM downlink messages

lteEmmUl	• 0 - do not report • 1 - report LTE EMM uplink messages
lteEmmDl	• 0 - do not report • 1 - report GSM/UMTS uplink messages
gsmUmtsUl	• 0 - do not report • 1 - report GSM/UMTS uplink messages
gsmUmtsDl	• 0 - do not report • 1 - report GSM/UMTS downlink messages
pRankIndicator-Ind	• 0 - do not report • 1 - report Rank Indicator messages
pTimer	• 0 - do not report • 1 - report Timer Indicator messages

8.274.2 Field Documentation

8.274.2.1 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsDl

8.274.2.2 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsUl

8.274.2.3 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmDl

8.274.2.4 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmUl

8.274.2.5 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmDl

8.274.2.6 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmUl

8.274.2.7 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pRankIndicatorInd

8.274.2.8 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pTimer

8.275 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN * pWcdmaUARFCN
- nas_lteEARFCN * pLteEARFCN
- nas_ltePCI * pLtePCI

8.275.1 Detailed Description

Structure for ChannelLock pack.

Parameters

pWcdmaUARFCN	[Optional] • See nas_wcdmaUARFCN for more information
pLteEARFCN	[Optional] • See nas_lteEARFCN for more information
pLtePCI	[Optional] • See nas_ltePCI for more information

8.275.2 Field Documentation

8.275.2.1 nas_lteEARFCN∗ pack_nas_SLQSNASSwiSetChannelLock_t::pLteEARFCN

8.275.2.2 nas_ltePCI∗ pack_nas_SLQSNASSwiSetChannelLock_t::pLtePCI

8.275.2.3 nas_wcdmaUARFCN∗ pack_nas_SLQSNASSwiSetChannelLock_t::pWcdmaUARFCN

8.276 pack_nas_SLQSSetBandPreference_t Struct Reference

Data Fields

- uint64_t bandPref

8.276.1 Detailed Description

Provides information about the band preference.

Parameters

bandpreference	• Bit mask representing the band preference to be set. • Bit position meanings: – 0 - BC0_A - Band Class 0, A-System – 1 - BC0_B - Band Class 0, B-System, Band Class 0 AB , GSM 850 Band – 2 - BC1 - Band Class 1, all blocks – 3 - BC2 - Band Class 2 place holder – 4 - BC3 - Band Class 3, A-System – 5 - BC4 - Band Class 4, all blocks – 6 - BC5 - Band Class 5, all blocks – 7 - GSM_DCS_1800 - GSM DCS band – 8 - GSM_EGSM_900 - GSM Extended GSM (E-GSM) band – 9 - GSM_PGSM_900 - GSM Primary GSM (P-GSM) band – 10 - BC6 - Band Class 6 – 11 - BC7 - Band Class 7 – 12 - BC8 - Band Class 8 – 13 - BC9 - Band Class 9 – 14 - BC10 - Band Class 10 – 15 - BC11 - Band Class 11 – 16 - GSM_450 - GSM 450 band – 17 - GSM_480 - GSM 480 band – 18 - GSM_750 - GSM 750 band – 19 - GSM_850 - GSM 850 band – 20 - GSM_RGSM_900 - GSM Railways GSM Band – 21 - GSM_PCS_1900 - GSM PCS band – 22 - WCDMA_I_IMT_2000 - WCDMA EUROPE JAPAN and CHINA IMT 2100 band – 23 - WCDMA_II_PCS_1900 - WCDMA US PCS 1900 band – 24 - WCDMA_III_1700 - WCDMA EUROPE and CHINA DCS 1800 band – 25 - WCDMA_IV_1700 - WCDMA US 1700 band – 26 - WCDMA_V_850 - WCDMA US 850 band – 27 - WCDMA_VI_800 - WCDMA JAPAN 800 band – 28 - BC12 - Band Class 12 – 29 - BC14 - Band Class 14 – 30 - RESERVED_2 - Reserved 2 – 31 - BC15 - Band Class 15 – 32 - 47 - Reserved – 48 - WCDMA_VII_2600 - WCDMA EUROPE 2600 band – 49 - WCDMA_VIII_900 - WCDMA EUROPE and JAPAN 900 band – 50 - WCDMA_IX_1700 - WCDMA JAPAN 1700 band – 51 to 55 - Reserved – 56 - BBC16 - Band Class 16 – 57 - BC17 - Band Class 17 – 58 - BC18 - Band Class 18 – 59 - BC19 - Band Class 19 – 60 to 64 - Reserved

8.276.2 Field Documentation

8.276.2.1 uint64_t pack_nas_SLQSSetBandPreference_t::bandPref

8.277 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference

Data Fields

- uint8_t bEnable
- nas_SLQSSignalStrengthsIndReq * pSigIndReq

8.277.1 Detailed Description

Parameters

bEnable	0/1 to disable/enable RSSI signal stregth indication
pSigIndReq	parameters to control signal strength indication

8.277.2 Field Documentation

8.277.2.1 uint8_t pack_nas_SLQSSetSignalStrengthsCallback_t::bEnable

8.277.2.2 nas_SLQSSignalStrengthsIndReq* pack_nas_SLQSSetSignalStrengthsCallback_t::pSigIndReq

8.278 pack_nas_SLQSSetSysSelectionPref_t Struct Reference

Data Fields

- uint8_t * pEmerMode
- uint16_t * pModePref
- uint64_t * pBandPref
- uint16_t * pPRLPref
- uint16_t * pRoamPref
- uint64_t * pLTEBandPref
- struct nas_netSelectionPref * pNetSelPref
- uint8_t * pChgDuration
- uint8_t * pMNCIncPCSDigStat
- uint32_t * pSrvDomainPref
- uint32_t * pGWAcqOrderPref
- uint64_t * pTdscdmaBandPref
- struct nas_acqOrderPref * pAcqOrderPref
- uint32_t * pSrvRegRestriction
- struct nas_CSGID * pCSGID
- unsigned char * pRAT

8.278.1 Detailed Description

Contain the system selection preferences.

Parameters

pEmerMode	• Optional parameter specifying the emergency Mode • Values: – 0 - OFF (normal) – 1 - ON (Emergency)
pModePref	• Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE

pBandPref	• Optional parameter • Bit mask representing the band preference • Bit values: – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved

pPRLPref	• Optional parameter indicating the CDMA PRL Preference • Values: – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems
pRoamPref	• Optional parameter indicating the roaming Preference • Values: – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator
pLTEBandPref	• Optional parameter • Bit mask representing the LTE band preference • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – All other bits are reserved

pNetSelPref	- netSelectionPref • Optional parameter for specifying Network Selection Preference • Modem selects networks based on this parameter(if present). • see nas_netSelectionPref for more information
pChgDuration	• Optional parameter specifying the duration of the change • Values: – 0x00 - Power cycle - Remains active until the next device power cycle – 0x01 - Permanent - Remains active through power cycles until changed by client – Device will use "0x01 - permanent" as default if this parameter is omitted
pMNCIncPCS-DigStat	• Optional parameter indicating if MNC includes PCS digit • Values: – TRUE - MNC is a 3 digit value; e.g., a reported value of 90 corresponds to an MNC value of 090 – FALSE - MNC is a 2-digit value; e.g., a reported value of 90 corresponds to an MNC value of 90
pSrvDomainPref	• Optional parameter indicating Service domain preference • Values: – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach
pGWAcqOrder-Pref	• Optional parameter indicating GSM/WCDMA Acquisition order Preference • Values: – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM
pTdscdmaBand-Pref	• Optional parameter indicating bitmask representing the TD-SCDMA band preference to be set. • Values: – 0x01 - TD-SCDMA Band A – 0x02 - TD-SCDMA Band B – 0x04 - TD-SCDMA Band C – 0x08 - TD-SCDMA Band D – 0x10 - TD-SCDMA Band E – 0x20 - TD-SCDMA Band F – All other bits are reserved
pAcqOrderPref	- acqOrderPref • Optional parameter for specifying Acquisition Order Preference • see nas_acqOrderPref for more information

pSrvReg-Restriction	• Optional parameter indicating Network Selection Registration Restriction Preference • Values: – 0x00 - Device follows the normal registration process – 0x01 - Device camps on the network according to its provisioning, but does not register – 0x02 - Device selects the network for limited service – All other values are reserved.
pCSGID	- CSGID • Optional parameter for specifying CSG ID • Either of pNetSelPref or pCSGID can be set. • see nas_CSGID for more information
pRAT	• Optional parameter Radio Access Technology order Preference • Values: – 0x04 - GSM – 0x05 - UMTS – 0x08 - LTE – 0x09 - TDSCDMA

8.278.2 Field Documentation

8.278.2.1 struct nas_acqOrderPref* pack_nas_SLQSSetSysSelectionPref_t::pAcqOrderPref

8.278.2.2 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pBandPref

8.278.2.3 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pChgDuration

8.278.2.4 struct nas_CSGID* pack_nas_SLQSSetSysSelectionPref_t::pCSGID

8.278.2.5 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pEmerMode

8.278.2.6 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pGWAcqOrderPref

8.278.2.7 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pLTEBandPref

8.278.2.8 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pMNCIncPCSDigStat

8.278.2.9 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pModePref

8.278.2.10 struct nas_netSelectionPref* pack_nas_SLQSSetSysSelectionPref_t::pNetSelPref

8.278.2.11 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pPRLPref

8.278.2.12 unsigned char* pack_nas_SLQSSetSysSelectionPref_t::pRAT

8.278.2.13 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pRoamPref

8.278.2.14 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvDomainPref

8.278.2.15 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvRegRestriction

8.278.2.16 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pTdscdmaBandPref

8.279 pack_nas_SLQSSwiPSDetach_t Struct Reference

Data Fields

- uint8_t * pDetachAction

8.279.1 Detailed Description

Structure for PS connection detach pack.

Parameters

pDetachAction[IN]	• Values – 2- Initiates an immediate packet domain detach.

8.279.2 Field Documentation

8.279.2.1 uint8_t* pack_nas_SLQSSwiPSDetach_t::pDetachAction

8.280 pack_pds_PDSInjectTimeReference_t Struct Reference

Data Fields

- uint64_t systemTime
- uint16_t systemDiscontinuities

8.280.1 Detailed Description

Structure contain the parameter for PDS Inject time.

Parameters

systemTime	• System time(milliseconds)
systemDiscontinuities	• Number of system time discontinuities

8.280.2 Field Documentation

8.280.2.1 uint16_t pack_pds_PDSInjectTimeReference_t::systemDiscontinuities

8.280.2.2 uint64_t pack_pds_PDSInjectTimeReference_t::systemTime

8.281 pack_pds_ResetPDSData_t Struct Reference

Data Fields

- uint32_t * pGPSDataMask
- uint32_t * pCellDataMask

8.281.1 Detailed Description

Structure contain the parameter to Resets the specified PDS data.

Parameters

pGPSDataMask	• Bitmask of GPS data to clear (optional) – 0x00000001 - EPH – 0x00000002 - ALM – 0x00000004 - POS – 0x00000008 - TIME – 0x00000010 - IONO – 0x00000020 - UTC – 0x00000040 - HEALTH – 0x00000080 - SVDIR – 0x00000100 - SVSTEER – 0x00000200 - SADATA – 0x00000400 - RTI – 0x00000800 - ALM_CORR – 0x00001000 - FREQ_BIAS_EST
pCellDataMask	• Bitmask of cell data to clear (optional) – 0x00000001 - POS – 0x00000002 - LATEST_GPS_POS – 0x00000004 - OTA_POS – 0x00000008 - EXT_REF_POS – 0x00000010 - TIMETAG – 0x00000020 - CELLID – 0x00000040 - CACHED_CELLID – 0x00000080 - LAST_SRV_CELL – 0x00000100 - CUR_SRV_CELL – 0x00000200 - NEIGHBOR_INFO

8.281.2 Field Documentation

8.281.2.1 uint32_t* pack_pds_ResetPDSData_t::pCellDataMask

8.281.2.2 uint32_t* pack_pds_ResetPDSData_t::pGPSDataMask

8.282 pack_pds_SetEventReportCallback_t Struct Reference

Data Fields

- uint8_t posDataNmea
- uint8_t rptPosData

8.282.1 Detailed Description

Structure contain the parameter for the automatic tracking configuration for the NMEA COM port.

Parameters

posDataNmea	• Report new position data in NMEA format. • Values – 0x00 - Disable – 0x01 - Enable
rptPosData	• Report new position data in raw format. • Values – 0x00 - Disable – 0x01 - Enable

8.282.2 Field Documentation

8.282.2.1 uint8_t pack_pds_SetEventReportCallback_t::posDataNmea

8.282.2.2 uint8_t pack_pds_SetEventReportCallback_t::rptPosData

8.283 pack_pds_SetPDSDefaults_t Struct Reference

Data Fields

- uint32_t operation
- uint8_t timeout
- uint32_t interval
- uint32_t accuracy

8.283.1 Detailed Description

Structure containing session configuration parameters.

Parameters

operation	• Current session operating mode – 0 - Standalone – 1 - MS based – 2 - MS assisted
timeout	• Maximum amount of time (seconds) to work on each fix, maximum is 255

interval	• Interval (seconds) between fix requests
accuracy	• Preferred accuracy threshold (meters)

8.283.2 Field Documentation

8.283.2.1 uint32_t pack_pds_SetPDSDefaults_t::accuracy

8.283.2.2 uint32_t pack_pds_SetPDSDefaults_t::interval

8.283.2.3 uint32_t pack_pds_SetPDSDefaults_t::operation

8.283.2.4 uint8_t pack_pds_SetPDSDefaults_t::timeout

8.284 pack_pds_SetPDSState_t Struct Reference

Data Fields

- uint32_t enable

8.284.1 Detailed Description

Structure contain PDS state parameters.

Parameters

enable	• Desired PDS state – Zero - disable – Non-Zero - enable

8.284.2 Field Documentation

8.284.2.1 uint32_t pack_pds_SetPDSState_t::enable

8.285 pack_pds_SetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t bAuto

8.285.1 Detailed Description

Structure contain the parameter for the automatic tracking configuration for the NMEA COM port.

Parameters

bAuto	• Enable automatic tracking for NMEA COM port – 0x00 - Disabled – 0x01 - Enabled

8.285.2 Field Documentation

8.285.2.1 uint32_t pack_pds_SetPortAutomaticTracking_t::bAuto

8.286 pack_pds_SetServiceAutomaticTracking_t Struct Reference

Data Fields

- uint32_t bAuto

8.286.1 Detailed Description

Structure contain the parameter to Sets the automatic tracking state.

Parameters

bAuto	• Automatic tracking session started for service – 0x00 - Disabled – 0x01 - Enabled

8.286.2 Field Documentation

8.286.2.1 uint32_t pack_pds_SetServiceAutomaticTracking_t::bAuto

8.287 pack_pds_SetXTRAAutomaticDownload_t Struct Reference

Data Fields

- uint32_t bEnabled
- uint16_t interval

8.287.1 Detailed Description

Structure contain the parameter to set the XTRA automatic database download configuration.

Parameters

bEnabled	• Automatic XTRA download status – 0 - Disabled – 1 - Enabled

interval	• Interval (hours) between XTRA downloads

8.287.2 Field Documentation

8.287.2.1 uint32_t pack_pds_SetXTRAAutomaticDownload_t::bEnabled

8.287.2.2 uint16_t pack_pds_SetXTRAAutomaticDownload_t::interval

8.288 pack_pds_SetXTRANetwork_t Struct Reference

Data Fields

- uint32_t preference

8.288.1 Detailed Description

Structure contain the parameter to Sets the XTRA WWAN network preference

Parameters

preference	• XTRA WWAN network preference – 0x00 - None (any available network) – 0x01 - Home-only, only when on home systems – 0x02 - Roam-only, only when on non-home systems

8.288.2 Field Documentation

8.288.2.1 uint32_t pack_pds_SetXTRANetwork_t::preference

8.289 pack_pds_SLQSGetAGPSConfig_t Struct Reference

Data Fields

- uint8_t * pNetworkMode

8.289.1 Detailed Description

Structure contain the parameter to PDS AGPS (MS-based) configuration pack.

Parameters

pNetworkMode	• Network Mode of AGPS Server [optional - should be present in Multimode Systems] – 0x00 - UMTS – 0x01 - CDMA

8.289.2 Field Documentation

8.289.2.1 uint8_t* pack_pds_SLQSGetAGPSConfig_t::pNetworkMode

8.290 pack_pds_SLQSPDSInjectAbsoluteTimeReference_t Struct Reference

Data Fields

- uint64_t timeMsec
- uint32_t timeUncMsec
- uint8_t timeBase
- uint8_t forceFlag

8.290.1 Detailed Description

Structure contain the parameter to Injects a absolute time reference.

Parameters

timeMsec	• Represents the number of milliseconds elapsed since either a GPS or UTC time base. If the time base is UTC, this value should NOT include leap seconds
timeUncMsec	• Time uncertainty in milliseconds
timeBase	• Time base – 0x00 - GPS (midnight, Jan 6, 1980) – 0x01 - UTC (midnight, Jan 1, 1970)
forceFlag	• Force acceptance of data

8.290.2 Field Documentation

8.290.2.1 uint8_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::forceFlag

8.290.2.2 uint8_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeBase

8.290.2.3 uint64_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeMsec

8.290.2.4 uint32_t pack_pds_SLQSPDSInjectAbsoluteTimeReference_t::timeUncMsec

8.291 pack_pds_SLQSPDSInjectPositionData_t Struct Reference

Data Fields

- uint64_t * pTimeStamp
- uint64_t * pLatitude
- uint64_t * pLongitude
- uint32_t * pAltitudeWrtEllipsoid

- uint32_t * pAltitudeWrtSealevel
- uint32_t * pHorizontalUncCircular
- uint32_t * pVerticalUnc
- uint8_t * pHorizontalConfidence
- uint8_t * pVerticalConfidence
- uint8_t * pPositionSource
- uint8_t * pTimeType

8.291.1 Detailed Description

Position Data Parameters from the external source to be injected to PDS engine.

Parameters

pTimeStamp	• Timestamp of the injected position in msec. The time can be of type UTC, GPS, or Age and is defined in the pTimeType parameter. If the pTimeType is not present, the timestamp shall be assumed to be UTC time
pLatitude	• Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. Value (in decimal degrees) in the range from -90 degrees to +90 degrees.Value in double float format (refer toIEEE Std 754-1985)
pLongitude	• Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of Greenwich meridian. Value (in decimal degrees) in the range from -180 degrees to +180 degrees.
pAltitudeWrt-Ellipsoid	• Height above the WGS-84 reference ellipsoid. Value conveys height (in meters). When injecting altitude information, the control point should include either this parameter or the pAltitudeWrt-Sealevel parameter. Value in single float format (refer to IEEE Std 754-1985)
pAltitudeWrt-Sealevel	• Height of MS above the mean sea level in units (in meters). When injecting altitude information, the control point should include either this parameter or the pAltitudeWrtEllipsoid parameter. Value in single float format (refer to IEEE Std 754-1985)
pHorizontalUnc-Circular	• Circular horizontal uncertainty (in meters). This parameter must be included if the latitude and longitude parameters are specified. Value in single float format (refer to IEEE Std 754-1985)
pVerticalUnc	• Vertical uncertainty (in meters). This parameter must be included if one of the altitude parameter are specified.Value in single float format (refer to IEEE Std 754-1985)
pHorizontal-Confidence	• Confidence value of the location horizontal uncertainty, specified as percentage, 1 to 100. This parameter must be included if the latitude and longitude parameters are specified.
pVertical-Confidence	• Confidence value of the location vertical uncertainty, specified as percentage, 1 to 100. This parameter must be included if one of the altitude paramters are specified.

pPositionSource	• Source of injected position: – 0x00 - Unknown – 0x01 - GPS – 0x02 - Cell ID – 0x03 - Enhanced cell ID – 0x04 - WiFi – 0x05 - Terrestrial – 0x06 - Terrestrial hybrid – 0x07 - Other
pTimeType	• Defines the time value set in the pTimeStamp parameter. – 0x00 - UTC Time: starting Jan 1, 1970 – 0x01 - GPS Time: starting Jan 6, 1980 – 0x02 - Age: Age of position information

8.291.2 Field Documentation

8.291.2.1 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pAltitudeWrtEllipsoid

8.291.2.2 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pAltitudeWrtSealevel

8.291.2.3 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pHorizontalConfidence

8.291.2.4 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pHorizontalUncCircular

8.291.2.5 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pLatitude

8.291.2.6 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pLongitude

8.291.2.7 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pPositionSource

8.291.2.8 uint64_t* pack_pds_SLQSPDSInjectPositionData_t::pTimeStamp

8.291.2.9 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pTimeType

8.291.2.10 uint8_t* pack_pds_SLQSPDSInjectPositionData_t::pVerticalConfidence

8.291.2.11 uint32_t* pack_pds_SLQSPDSInjectPositionData_t::pVerticalUnc

8.292 pack_pds_SLQSSetAGPSConfig_t Struct Reference

Data Fields

- uint32_t * pServerAddress
- uint32_t * pServerPort
- uint8_t * pServerURL
- uint8_t * pServerURLLength
- uint8_t * pNetworkMode

8.292.1 Detailed Description

Structure contain the parameter to Sets the PDS AGPS (MS-based) configuration.

Parameters

pServerAddress	• IPv4 address of AGPS server [optional]
pServerPort	• Port number of AGPS server [optional - should be present when pServerAddress is present]
pServerURL	• URL of the AGPS server [optional]
pServerURL-Length	• URL length of AGPS server [optional - should be present when pServerURL is present]
pNetworkMode	• Network Mode of AGPS Server [optional - should be present in Multimode Systems] – 0x00 - UMTS – 0x01 - CDMA

8.292.2 Field Documentation

8.292.2.1 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pNetworkMode

8.292.2.2 uint32_t* pack_pds_SLQSSetAGPSConfig_t::pServerAddress

8.292.2.3 uint32_t* pack_pds_SLQSSetAGPSConfig_t::pServerPort

8.292.2.4 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pServerURL

8.292.2.5 uint8_t* pack_pds_SLQSSetAGPSConfig_t::pServerURLLength

8.293 pack_pds_SLQSSetPositionMethodState_t Struct Reference

Data Fields

- uint8_t * pXtraTimeState
- uint8_t * pXtraDataState
- uint8_t * pWifiState

8.293.1 Detailed Description

Parameters to Set state of positioning method for a device.

Parameters

pXtraTimeState	• XTRA Time Position Method State. • Values: – 0x00 - Disable – 0x01 - Enable
pXtraDataState	• XTRA Data Position Method State. • Values: – 0x00 - Disable – 0x01 - Enable
Latitude	• WiFi Position Method State • Values: – 0x00 - Disable – 0x01 - Enable

8.293.2 Field Documentation

8.293.2.1 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pWifiState

8.293.2.2 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pXtraDataState

8.293.2.3 uint8_t* pack_pds_SLQSSetPositionMethodState_t::pXtraTimeState

8.294 pack_pds_StartPDSTrackingSessionExt_t Struct Reference

Data Fields

- uint8_t sessionControl
- uint8_t sessionType
- uint8_t sessionOperation
- uint8_t sessionServerOption
- uint8_t fixTimeout
- uint32_t fixInterval
- uint32_t fixCount
- uint32_t fixAccuracy

8.294.1 Detailed Description

Structure contain the parameter for PDS tracking session.

Parameters

sessionControl	• Control method: – 0x0 - Manual

sessionType	• Type: – 0x0 - New
session-Operation	• Operating mode: – 0x00 - Standalone – 0x01 - MS-based
sessionServer-Option	• Location server option: – 0x0 - Default
fixTimeout	• Maximum time to work on each fix (in seconds, max 255)
fixCount	• Count of position fix requests for this session (must be at least 1)
fixInterval	• interval between position fix requests (in seconds)
fixAccuracy	• Preferred accuracy threshold(in meters)

8.294.2 Field Documentation

8.294.2.1 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixAccuracy

8.294.2.2 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixCount

8.294.2.3 uint32_t pack_pds_StartPDSTrackingSessionExt_t::fixInterval

8.294.2.4 uint8_t pack_pds_StartPDSTrackingSessionExt_t::fixTimeout

8.294.2.5 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionControl

8.294.2.6 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionOperation

8.294.2.7 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionServerOption

8.294.2.8 uint8_t pack_pds_StartPDSTrackingSessionExt_t::sessionType

8.295 pack_qmi_t Struct Reference

Data Fields

- uint16_t xid
- int timeout
- uint16_t msgid
- uint8_t svc

8.295.1 Detailed Description

qmi request context

Parameters

in	*xid*	transaction id
out	*timeout*	timeout recommended in seconds
out	*msgid*	message id
out	*svc*	qmi service

8.295.2 Field Documentation

8.295.2.1 uint16_t pack_qmi_t::msgid

8.295.2.2 uint8_t pack_qmi_t::svc

8.295.2.3 int pack_qmi_t::timeout

8.295.2.4 uint16_t pack_qmi_t::xid

8.296 pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference

Data Fields

- uint32_t apnId

8.296.1 Detailed Description

Structure that contains the APN ID to obtain extra APN parameters

Parameters

apnId[IN]	• APN id

8.296.2 Field Documentation

8.296.2.1 uint32_t pack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.297 pack_qos_SLQSQosSwiReadDataStats_t Struct Reference

Data Fields

- uint32_t apnId

8.297.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

apnId[IN]	• APN id

8.297.2 Field Documentation

8.297.2.1 uint32_t pack_qos_SLQSQosSwiReadDataStats_t::apnId

8.298 pack_qos_SLQSSetQosEventCallback_t Struct Reference

Data Fields

- uint8_t enable

8.298.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

`in`	*enable*	• 1 - Enable QoS event reporting • 0 - Disable QoS event reporting

8.298.2 Field Documentation

8.298.2.1 uint8_t pack_qos_SLQSSetQosEventCallback_t::enable

8.299 pack_rms_SetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask

8.299.1 Detailed Description

Parameters

enabled	• SMS wake functionality enabled – 0 - Disabled – 1 - Enabled
wake_mask	• SMS wake mask to search for incoming messages (only relevant when enabled)

8.299.2 Field Documentation

8.299.2.1 uint32_t pack_rms_SetSMSWake_t::enabled

8.299.2.2 uint32_t pack_rms_SetSMSWake_t::wake_mask

8.300 pack_sar_SLQSSetRfSarState_t Struct Reference

Data Fields

- uint32_t RfSarState

8.300.1 Detailed Description

Parameters

RfSarState	• SAR RF State – QMI_SAR_RF_STATE_DEFAULT – QMI_SAR_RF_STATE_1 – QMI_SAR_RF_STATE_2 – QMI_SAR_RF_STATE_3 – QMI_SAR_RF_STATE_4 – QMI_SAR_RF_STATE_5 – QMI_SAR_RF_STATE_6 – QMI_SAR_RF_STATE_7 – QMI_SAR_RF_STATE_8

8.300.2 Field Documentation

8.300.2.1 uint32_t pack_sar_SLQSSetRfSarState_t::RfSarState

8.301 pack_sms_SaveSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage

8.301.1 Detailed Description

Sutructure containing parameters to Save an SMS message to device memory

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV

messageFormat	• Message format – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
messageSize	• The length of the message contents in bytes
pMessage	• The message contents

8.301.2 Field Documentation

8.301.2.1 uint32_t pack_sms_SaveSMS_t::messageFormat

8.301.2.2 uint32_t pack_sms_SaveSMS_t::messageSize

8.301.2.3 uint8_t* pack_sms_SaveSMS_t::pMessage

8.301.2.4 uint32_t pack_sms_SaveSMS_t::storageType

8.302 pack_sms_SendSMS_t Struct Reference

Data Fields

- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage
- uint8_t * pLinktimer

8.302.1 Detailed Description

Parameters

messageFormat	• Message format – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
messageSize	• The length of the message contents in bytes
pLinktimer	• GW SMS link open for the specified number of second
pMessage	• The message contents in PDU format contains SMS header and payload message

8.302.2 Field Documentation

8.302.2.1 uint32_t pack_sms_SendSMS_t::messageFormat

8.302.2.2 uint32_t pack_sms_SendSMS_t::messageSize

8.302.2.3 uint8_t* pack_sms_SendSMS_t::pLinktimer

8.302.2.4 uint8_t* pack_sms_SendSMS_t::pMessage

8.303 pack_sms_SetNewSMSCallback_t Struct Reference

Data Fields

- enum eqmiCbkSetStatus status

8.303.1 Detailed Description

Parameters

status	callback parameter

8.303.2 Field Documentation

8.303.2.1 enum eqmiCbkSetStatus pack_sms_SetNewSMSCallback_t::status

8.304 pack_sms_SetSMSCAddress_t Struct Reference

Data Fields

- uint8_t * pSMSCAddress
- uint8_t * pSMSCType

8.304.1 Detailed Description

Structure to Set the SMS center address.

Parameters

SMSCAddress	• The SMS center address represented as a NULL terminated string
SMSCType	• The SMS center address type represented as a NULL terminated string (optional).

8.304.2 Field Documentation

8.304.2.1 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCAddress

8.304.2.2 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCType

8.305 pack_sms_SLQSDeleteSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t * pMessageIndex
- uint32_t * pMessageTag
- uint8_t * pMessageMode

8.305.1 Detailed Description

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
pMessageIndex	• (Optional) message index
pMessageTag	• (Optional) message tag – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
pMessageMode	• (Optional) message mode • this must be included if the device is capable of supporting more than one protocol • e.g. CDMA and GW – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)

8.305.2 Field Documentation

8.305.2.1 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageIndex

8.305.2.2 uint8_t* pack_sms_SLQSDeleteSMS_t::pMessageMode

8.305.2.3 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageTag

8.305.2.4 uint32_t pack_sms_SLQSDeleteSMS_t::storageType

8.306 pack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex
- uint8_t * pMessageMode

8.306.1 Detailed Description

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
messageIndex	• Message index
pMessageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.306.2 Field Documentation

8.306.2.1 uint32_t pack_sms_SLQSGetSMS_t::messageIndex

8.306.2.2 uint8_t* pack_sms_SLQSGetSMS_t::pMessageMode

8.306.2.3 uint32_t pack_sms_SLQSGetSMS_t::storageType

8.307 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode

8.307.1 Detailed Description

Parameters

mode	• Mode – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)

8.307.2 Field Documentation

8.307.2.1 uint8_t pack_sms_SLQSGetSmsBroadcastConfig_t::mode

8.308 pack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t * pRequestedTag
- uint8_t * pMessageMode

8.308.1 Detailed Description

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
requestedTag	• (Optional) Message tag – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
messageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.308.2 Field Documentation

8.308.2.1 uint8_t* pack_sms_SLQSGetSMSList_t::pMessageMode

8.308.2.2 uint32_t* pack_sms_SLQSGetSMSList_t::pRequestedTag

8.308.2.3 uint32_t pack_sms_SLQSGetSMSList_t::storageType

8.309 pack_sms_SLQSModifySMSStatus_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex
- uint32_t messageTag
- uint8_t * pMessageMode

8.309.1 Detailed Description

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
messageIndex	• Message index

messageTag	• Message tag – 0 - Read – 1 - Not read
pMessageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.309.2 Field Documentation

8.309.2.1 uint32_t pack_sms_SLQSModifySMSStatus_t::messageIndex

8.309.2.2 uint32_t pack_sms_SLQSModifySMSStatus_t::messageTag

8.309.2.3 uint8_t* pack_sms_SLQSModifySMSStatus_t::pMessageMode

8.309.2.4 uint32_t pack_sms_SLQSModifySMSStatus_t::storageType

8.310 pack_sms_SLQSSendAsyncSMS_t Struct Reference

Data Fields

- sms_sendAsyncsmsParams * pSendSmsParams

8.310.1 Detailed Description

Structure contain Parameter to Send an SMS message for immediate over-the-air transmission

Parameters

pSendSms-Params	• structure containing the SMS parameters. Refer sms_sendasyncsmsparams

8.310.2 Field Documentation

8.310.2.1 sms_sendAsyncsmsParams* pack_sms_SLQSSendAsyncSMS_t::pSendSmsParams

8.311 pack_sms_SLQSSetIndicationRegister_t Struct Reference

Data Fields

- sms_setIndicationReg * pSetIndicationRegReq

8.311.1 Detailed Description

Structure contain Parameter that sets the registration state of different WMS indications.

Parameters

pSetIndication-Reg	• Pointer to structure of indicationRegReqParams – See sms_setIndicationReg for more information

8.311.2 Field Documentation

8.311.2.1 sms_setIndicationReg∗ pack_sms_SLQSSetIndicationRegister_t::pSetIndicationRegReq

8.312 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference

Data Fields

- uint8_t mode
- uint8_t broadcastActivate

8.312.1 Detailed Description

Structure contain parameters that Enables or disables the reception of broadcast SMS messages.

Parameters

Mode	• Mode • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)
broadcast-Activate	• 0x00 - Disable broadcast • 0x01 - Activate broadcast

8.312.2 Field Documentation

8.312.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::broadcastActivate

8.312.2.2 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::mode

8.313 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode
- sms_qaQmi3GPPBroadcastCfgInfo ∗ pBroadcastConfig
- sms_qaQmi3GPP2BroadcastCfgInfo ∗ pCDMABroadcastConfig

8.313.1 Detailed Description

Structure containing parameters that provides the information about the SMS BroadcastConfiguration

Parameters

mode	• Mode – 0x00 - CDMA, LTE (if network type is CDMA) – 0x01 - GW, LTE (if network type is UMTS)
pBroadcast-Config	• The data for 3GPP Broadcast Information(Optional).
pCDMA-BroadcastConfig	• The data for 3GPP2 Broadcast Information(Optional).

8.313.2 Field Documentation

8.313.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastConfig_t::mode

8.313.2.2 sms_qaQmi3GPPBroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pBroadcastConfig

8.313.2.3 sms_qaQmi3GPP2BroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pCDMABroadcastConfig

8.314 pack_sms_SLQSSetSmsStorage_t Struct Reference

Data Fields

- uint8_t smsStorage

8.314.1 Detailed Description

Structure contain Parameter to set the SMS Storage on the device

Parameters

smsStorage	• SMS Storage – 0x01 - device's permanent memory – 0x02 - UICC

8.314.2 Field Documentation

8.314.2.1 uint8_t pack_sms_SLQSSetSmsStorage_t::smsStorage

8.315 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeReq * pMaxStorageSizeReq

8.315.1 Detailed Description

Parameters

pMaxStorage-SizeReq	• Request parameters for SmsSLQSGetMaxStorageSize – See sms_maxStorageSizeReq for more information

8.315.2 Field Documentation

8.315.2.1 sms_maxStorageSizeReq∗ pack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeReq

8.316 pack_sms_SLQSSmsSetRoutes_t Struct Reference

Data Fields

- sms_setRoutesReq ∗ pSetRoutesReq

8.316.1 Detailed Description

Structure contain parameter that Sets the action performed on SMS message receipt for specified message routes. It also specifies the action performed on SMS receipt of status reports.

Parameters

pSetRoutesReq	• Pointer to structure of smsSetRoutesReq – See sms_setRoutesReq for more information

8.316.2 Field Documentation

8.316.2.1 sms_setRoutesReq∗ pack_sms_SLQSSmsSetRoutes_t::pSetRoutesReq

8.317 pack_swiavms_SLQSAVMSSendSelection_t Struct Reference

Data Fields

- uint8_t selection
- uint32_t ∗ pDeferTime
- uint8_t ∗ pClientPerformOperationFlag
- uint8_t ∗ pPackageID
- uint16_t ∗ pRejectReason

8.317.1 Detailed Description

Structure containing the AVMS selection

Parameters

selection[IN]	• User Selection – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer
pDeferTime[IN/-OPTIONAL]	• Defer time in minutes. A value of 0 will cause the prompt to be resent immediately.
pClientPerform-OperationFlag[I-N/OPTIONAL]	• Client operation flag after accept. – 0: if modem performs the operation (download or update) – 1: if client performs the operation (download or update)
pPackageID[IN/-OPTIONAL]	• Package ID.
pRejectReason[-IN/OPTIONAL]	• Reject Reason.

8.317.2 Field Documentation

8.317.2.1 uint8_t* pack_swiavms_SLQSAVMSSendSelection_t::pClientPerformOperationFlag

8.317.2.2 uint32_t* pack_swiavms_SLQSAVMSSendSelection_t::pDeferTime

8.317.2.3 uint8_t* pack_swiavms_SLQSAVMSSendSelection_t::pPackageID

8.317.2.4 uint16_t* pack_swiavms_SLQSAVMSSendSelection_t::pRejectReason

8.317.2.5 uint8_t pack_swiavms_SLQSAVMSSendSelection_t::selection

8.318 pack_swiavms_SLQSAVMSSetSettings_t Struct Reference

Data Fields

- uint8_t FwAutodownload
- uint8_t FwAutoUpdate
- uint8_t AutoConnect
- uint8_t AutoReboot
- uint8_t * pFwAutoSDM
- uint32_t * pPollingTimer
- PackSwiAvmsSetSettingsConnectionRetryTimers * pConnectionRetryTimers
- PackSwiAvmsSetSettingsAPNInfo * pAPNInfo
- uint8_t * pNotifStore
- PackSwiAvmsSetSettingsPeriodInfo * pPeriodInfo

8.318.1 Detailed Description

Structure containing the AVMS settings to be set on the device This maps to structure SLQSAVMSSetSettings

Parameters

FwAutodownload[IN]	• Firmware Auto Download – 0x00 - FALSE – 0x01 - TRUE
Firmware	Auto Update[IN] • 1 byte parameter indicating FOTA Automatic update – 0x00 - Firmware autoupdate FALSE – 0x01 - Firmware autoupdate TRUE
AutoConnect[IN]	• Auto Connect – 0x00 - FALSE – 0x01 - TRUE
AutoReboot[IN]	• Automatic device reboot when the request is received from AirVantage server – 0x00 - FALSE – 0x01 - TRUE
pFwAutoSDM[IN/OPTIONAL]	• OMA Automatic UI Alert Response – 0x00 - DISABLED – 0x01 - ENABLED ACCEPT – 0x02 - ENABLED REJECT
pPollingTimer[IN/OPTIONAL]	• Polling timer to connect to AVMS server – 0-525600 (min) – 0:disabled
pConnectionRetryTimers[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsConnectionRetryTimers for more information
pAPNInfo[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information
pNotifStore[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information
pPeriodInfo[IN/OPTIONAL]	• See PackSwiAvmsSetSettingsAPNInfo for more information

8.318.2 Field Documentation

8.318.2.1 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::AutoConnect

8.318.2.2 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::AutoReboot

8.318.2.3 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::FwAutodownload

8.318.2.4 uint8_t pack_swiavms_SLQSAVMSSetSettings_t::FwAutoUpdate

8.318.2.5 PackSwiAvmsSetSettingsAPNInfo* pack_swiavms_SLQSAVMSSetSettings_t::pAPNInfo

8.318.2.6 PackSwiAvmsSetSettingsConnectionRetryTimers* pack_swiavms_SLQSAVMSSetSettings_t::pConnectionRetryTimers

8.318.2.7 uint8_t* pack_swiavms_SLQSAVMSSetSettings_t::pFwAutoSDM

8.318.2.8 uint8_t* pack_swiavms_SLQSAVMSSetSettings_t::pNotifStore

8.318.2.9 PackSwiAvmsSetSettingsPeriodInfo* pack_swiavms_SLQSAVMSSetSettings_t::pPeriodInfo

8.318.2.10 uint32_t* pack_swiavms_SLQSAVMSSetSettings_t::pPollingTimer

8.319 pack_swiavms_SLQSAVMSStartSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.319.1 Detailed Description

Structure that contains the session type request for AVMS start session command

Parameters

sessionType[IN]	• Session type – 0x01 - FOTA, to check availability of FW Update. This field is mandatory in OMA-DM case but not necessary in LWM2M one. To keep compatibility, this shall be used in LWM2M without any specific treatment.

8.319.2 Field Documentation

8.319.2.1 uint8_t pack_swiavms_SLQSAVMSStartSession_t::sessionType

8.320 pack_swiavms_SLQSAVMSStopSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.320.1 Detailed Description

Structure that contains the session type for AVMS Stop session command

Parameters

sessionType	• Session type – 0x01 - FOTA, to suspend FOTA session. – 0xFF - Suspend ongoing FOTA session or stop any other active AVMS session.
resultcode	• Tlv Result Code

8.320.2 Field Documentation

8.320.2.1 uint8_t pack_swiavms_SLQSAVMSStopSession_t::sessionType

8.321 pack_swiloc_SwiLocSetAutoStart_t Struct Reference

Data Fields

- uint8_t function
- int set_function
- uint8_t fix_type
- int set_fix_type
- uint8_t max_time
- int set_max_time
- uint32_t max_dist
- int set_max_dist
- uint32_t fix_rate
- int set_fix_rate

8.321.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

function	• Setting to indicate when modem should start an automatic GNSS fix – 0 - disabled – 1 - At bootup – 2 - When NMEA port is opened
set_function	• 0 - do not set to modem • I - set to modem
fix_type	• Type of GNSS fix: – 1 - Default Engine mode – 2 - MS-Based – 3 - MS-Assisted – 4 - Standalone

set_fix_type	• 0 - do not set to modem • l - set to modem
max_time	• Maximum time allowed for the receiver to get a fix in seconds • Valid range: 1-255
set_max_time	• 0 - do not set to modem • l - set to modem
max_dist	• Maximum uncertainty of a fix measured by distance in meters • Valid range: 1 - 4294967280
set_max_dist	• 0 - do not set to modem • l - set to modem
fix_rate	• Time between fixes in seconds • Valid range: 1–65535
set_fix_rate	• 0 - do not set to modem • l - set to modem

8.321.2 Field Documentation

8.321.2.1 uint32_t pack_swiloc_SwiLocSetAutoStart_t::fix_rate

8.321.2.2 uint8_t pack_swiloc_SwiLocSetAutoStart_t::fix_type

8.321.2.3 uint8_t pack_swiloc_SwiLocSetAutoStart_t::function

8.321.2.4 uint32_t pack_swiloc_SwiLocSetAutoStart_t::max_dist

8.321.2.5 uint8_t pack_swiloc_SwiLocSetAutoStart_t::max_time

8.321.2.6 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_rate

8.321.2.7 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_type

8.321.2.8 int pack_swiloc_SwiLocSetAutoStart_t::set_function

8.321.2.9 int pack_swiloc_SwiLocSetAutoStart_t::set_max_dist

8.321.2.10 int pack_swiloc_SwiLocSetAutoStart_t::set_max_time

8.322 pack_swioma_SLQSOMADMCancelSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.322.1 Detailed Description

Structure that contains the session type for OMA cancel session command

Parameters

in	*sessionType*	• Session type – 0x01 - FOTA, to check availability of FW Update – 0xFF - Cancel any active OMADM session

8.322.2 Field Documentation

8.322.2.1 uint32_t pack_swioma_SLQSOMADMCancelSession_t::sessionType

8.323 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference

Data Fields

- uint32_t SessionType

8.323.1 Detailed Description

Structure that contains the session type for OMA get session info command

Parameters

in	*SessionType*	• Session type – 0x01 - FOTA – 0xFF - Any active OMADM session. If no active sessions are available, then previous OMADM session info is returned

8.323.2 Field Documentation

8.323.2.1 uint32_t pack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.324 pack_swioma_SLQSOMADMSendSelection_t Struct Reference

Data Fields

- uint32_t selection
- uint32_t * pDeferTime
- uint32_t * pRejectReason

8.324.1 Detailed Description

Structure containing the OMA DM selection

Parameters

in	*selection*	• OMA-DM NIA Selection – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer
in	*pDeferTime*	• Defer time in minutes. A value of 0 will cause the prompt to be resent immediately. • This TLV is mandatory if selection is set to 0x03.
in	*pRejectReason*	• Reject Reason • This TLV is processed if selection is set to 0x02. If it is not present, the reject reason 0 is used as default.

8.324.2 Field Documentation

8.324.2.1 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pDeferTime

8.324.2.2 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pRejectReason

8.324.2.3 uint32_t pack_swioma_SLQSOMADMSendSelection_t::selection

8.325 pack_swioma_SLQSOMADMSetSettings_t Struct Reference

Data Fields

- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t * pAutosdm
- uint8_t * pFwAutoCheck

8.325.1 Detailed Description

Structure containing the OMA DM settings to be set on the device This maps to structure SLQSOMADMSettings-ReqParams3

Parameters

FOTAdownload	• 1 Byte parameter indicating support for FOTA Automatic download – 0x00 - Firmware autodownload FALSE – 0x01 - Firmware autodownload TRUE

FOTAUpdate	• 1 byte parameter indicating FOTA Automatic update – 0x00 - Firmware autoupdate FALSE – 0x01 - Firmware autoupdate TRUE
pAutosdm	• Optional 1 byte parameter indicating OMA Automatic UI Alert Response – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject
pFwAutoCheck	• Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response – 0x00 - Disabled – 0x01 - Enabled

8.325.2 Field Documentation

8.325.2.1 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAdownload

8.325.2.2 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAUpdate

8.325.2.3 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pAutosdm

8.325.2.4 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pFwAutoCheck

8.326 pack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.326.1 Detailed Description

Structure that contains the session type for OMA start session command

Parameters

sessionType[IN]	• Session type – 0x01 - FOTA, to check availability of FW Update – 0x02 - DM, to check availability of DM Update – 0x03 - PRL, to check availability of PRL Update

8.326.2 Field Documentation

8.326.2.1 uint32_t pack_swioma_SLQSOMADMStartSession_t::sessionType

8.327 pack_uim_ChangePin_t Struct Reference

Data Fields

- uim_encryptedPIN1 EncryptedPIN1
- uint32_t * pIndicationToken
- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_changeUIMPIN changePIN
- uint16_t TlvResult

8.327.1 Detailed Description

This structure contains information of the request parameters associated with a Change PIN API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
changePIN	• See uim_changeUIMPIN for more information.
pKeyReferenceID(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.327.2 Field Documentation

8.327.2.1 uim_changeUIMPIN pack_uim_ChangePin_t::changePIN

8.327.2.2 uim_encryptedPIN1 pack_uim_ChangePin_t::EncryptedPIN1

8.327.2.3 uint32_t* pack_uim_ChangePin_t::pIndicationToken

8.327.2.4 uint8_t* pack_uim_ChangePin_t::pKeyReferenceID

8.327.2.5 uim_sessionInformation pack_uim_ChangePin_t::sessionInfo

8.327.2.6 uint16_t pack_uim_ChangePin_t::TlvResult

8.328 pack_uim_ReadTransparent_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_fileInfo fileIndex
- uim_readTransparentInfo readTransparent
- uint32_t * pIndicationToken
- uint8_t * pEncryptData
- uint16_t TlvResult

8.328.1 Detailed Description

This structure contains information of the request parameters associated with a Read Transparent API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
fileIndex	• See uim_fileInfo for more information.
readTransparent	• See uim_readTransparentInfo for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function
pEncrypt-Data(optional)	• Encrypt Data. • Indicates whether the data read from the card is to be encrypted.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.328.2 Field Documentation

8.328.2.1 uim_fileInfo pack_uim_ReadTransparent_t::fileIndex

8.328.2.2 uint8_t* pack_uim_ReadTransparent_t::pEncryptData

8.328.2.3 uint32_t* pack_uim_ReadTransparent_t::pIndicationToken

8.328.2.4 uim_readTransparentInfo pack_uim_ReadTransparent_t::readTransparent

8.328.2.5 uim_sessionInformation pack_uim_ReadTransparent_t::sessionInfo

8.328.2.6 uint16_t pack_uim_ReadTransparent_t::Tlvresult

8.329 pack_uim_SetPinProtection_t Struct Reference

Data Fields

- uim_encryptedPIN1 EncryptedPIN1
- uint32_t * pIndicationToken
- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_setPINProtection pinProtection
- uint16_t Tlvresult

8.329.1 Detailed Description

This structure contains information of the request parameters associated with a set pin protection API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
pinProtection	• See uim_setPINProtection for more information.
pKeyReferenceID(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.329.2 Field Documentation

8.329.2.1 uim_encryptedPIN1 pack_uim_SetPinProtection_t::EncryptedPIN1

8.329.2.2 uint32_t* pack_uim_SetPinProtection_t::pIndicationToken

8.329.2.3 uim_setPINProtection pack_uim_SetPinProtection_t::pinProtection

8.329.2.4 uint8_t* pack_uim_SetPinProtection_t::pKeyReferenceID

8.329.2.5 uim_sessionInformation pack_uim_SetPinProtection_t::sessionInfo

8.329.2.6 uint16_t pack_uim_SetPinProtection_t::Tlvresult

8.330 pack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_authenticationData authData
- uint32_t * pIndicationToken

8.330.1 Detailed Description

This structure contains information of the request parameters associated with a SLQSUIMAuthenticate.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
authData	• See uim_authenticationData for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.330.2 Field Documentation

8.330.2.1 uim_authenticationData pack_uim_SLQSUIMAuthenticate_t::authData

8.330.2.2 uint32_t* pack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.330.2.3 uim_sessionInformation pack_uim_SLQSUIMAuthenticate_t::sessionInfo

8.331 pack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_depersonalizationInformation depersonilisationInfo

8.331.1 Detailed Description

This structure contains information of the request parameters associated with a Depersonalization.

Parameters

depersonilisation-Info	• See uim_depersonalizationInformation for more information.

8.331.2 Field Documentation

8.331.2.1 uim_depersonalizationInformation pack_uim_SLQSUIMDepersonalization_t::depersonilisationInfo

8.332 pack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask

8.332.1 Detailed Description

Parameters

eventMask	- bit 1 - card status • bit 4 - physical slot status

8.332.2 Field Documentation

8.332.2.1 uint32_t pack_uim_SLQSUIMEventRegister_t::eventMask

8.333 pack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint32_t * pConfigurationMask

8.333.1 Detailed Description

This structure contains information of the request parameters associated with Get Configuration to get the modem configuration for the UIM module

Parameters

pConfiguration-Mask(optional)	• Requested configurations – Bit 0 - Automatic selection – Bit 1 - Personalization status – Bit 2 - Halt subscription – All other bits are reserved for future use

Note

- if the TLV is missing, the service returns all configuration items in the response.

8.333.2 Field Documentation

8.333.2.1 uint32_t* pack_uim_SLQSUIMGetConfiguration_t::pConfigurationMask

8.334 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_fileInfo fileIndex
- uint32_t * pIndicationToken

8.334.1 Detailed Description

This structure contains information of the request parameters associated with a Get File Attributes

Parameters

sessionInfo	• See uim_sessionInformation for more information.
fileIndex	• See uim_fileInfo for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.334.2 Field Documentation

8.334.2.1 uim_fileInfo pack_uim_SLQSUIMGetFileAttributes_t::fileIndex

8.334.2.2 uint32_t* pack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.334.2.3 uim_sessionInformation pack_uim_SLQSUIMGetFileAttributes_t::sessionInfo

8.335 pack_uim_SLQSUIMPowerDown_t Struct Reference

Data Fields

- uint8_t slot

8.335.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

slot	• Indicates the slot to be used. – 1 - Slot 1 – 2 - Slot 2

8.335.2 Field Documentation

8.335.2.1 uint8_t pack_uim_SLQSUIMPowerDown_t::slot

8.336 pack_uim_SLQSUIMPowerUp_t Struct Reference

Data Fields

- uint8_t slot
- uint8_t * pIgnoreHotSwapSwitch

8.336.1 Detailed Description

This structure contains information of the request parameters associated with a Power Down.

Parameters

slot	• Indicates the slot to be used. – 1 - Slot 1 – 2 - Slot 2
pIgnoreHot-Swap-Switch(optional)	• Hot-swap switch status. – 0 - Checks the hot-swap switch status – 1 - Ignores the hot-swap switch status

8.336.2 Field Documentation

8.336.2.1 uint8_t* pack_uim_SLQSUIMPowerUp_t::pIgnoreHotSwapSwitch

8.336.2.2 uint8_t pack_uim_SLQSUIMPowerUp_t::slot

8.337 pack_uim_SLQSUIMRefreshComplete_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t refreshComplete

8.337.1 Detailed Description

This structure contains information of the request parameters associated with a Refresh Complete event.

Parameters

sessionInfo(-Mandatory)	• See uim_sessionInformation for more information.
refreshComplete(-Mandatory)	• Indicates whether the refresh was successful. Valid values: – 0 - Refresh was not completed successfully – 1 - Refresh was completed successfully

8.337.2 Field Documentation

8.337.2.1 uint8_t pack_uim_SLQSUIMRefreshComplete_t::refreshComplete

8.337.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshComplete_t::sessionInfo

8.338 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo

8.338.1 Detailed Description

This structure contains information of the request parameters associated with a Refresh Get Last Event.

Parameters

sessionInfo(-Mandatory)	• See uim_sessionInformation for more information.

8.338.2 Field Documentation

8.338.2.1 uim_sessionInformation pack_uim_SLQSUIMRefreshGetLastEvent_t::sessionInfo

8.339 pack_uim_SLQSUIMRefreshOK_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t OKtoRefresh

8.339.1 Detailed Description

This structure contains Parameters of the Session Information

Parameters

sessionInfo	• Session Information • See uim_sessionInformation for more information
OKtoRefresh	• Indicates whether a refresh is OK. Valid values: – 0 - Not OK to refresh – 1 - OK to refresh

8.339.2 Field Documentation

8.339.2.1 uint8_t pack_uim_SLQSUIMRefreshOK_t::OKtoRefresh

8.339.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshOK_t::sessionInfo

8.340 pack_uim_SLQSUIMRefreshRegister_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_registerRefresh regRefresh

8.340.1 Detailed Description

This structure contains information of the request parameters associated with a Refresh Register.

Parameters

sessionInfo	• Session Information params • See uim_sessionInformation for more information
regRefresh	• Register Refresh parameters • See uim_registerRefresh for more information

8.340.2 Field Documentation

8.340.2.1 uim_registerRefresh pack_uim_SLQSUIMRefreshRegister_t::regRefresh

8.340.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshRegister_t::sessionInfo

8.341 pack_uim_SLQSUIMSwitchSlot_t Struct Reference

Data Fields

- uint8_t bLogicalSlot
- uint32_t ulPhysicalSlot

8.341.1 Detailed Description

This structure contains information of the request parameters associated with a Switch Slot.

Parameters

bLogicalSlot	• Indicates the slot to be used. – 1 - Slot 1 – 2 - Slot 2 – 3 - Slot 3 – 4 - Slot 4 – 5 - Slot 5
ulPhysicalSlot	• 1 - Slot 1 • 2 - Slot 2 • 3 - Slot 3 • 4 - Slot 4 • 5 - Slot 5

8.341.2 Field Documentation

8.341.2.1 uint8_t pack_uim_SLQSUIMSwitchSlot_t::bLogicalSlot

8.341.2.2 uint32_t pack_uim_SLQSUIMSwitchSlot_t::ulPhysicalSlot

8.342 pack_uim_UnblockPin_t Struct Reference

Data Fields

- uim_encryptedPIN1 EncryptedPIN1
- uint32_t * pIndicationToken
- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_unblockUIMPIN pinProtection
- uint16_t Tlvresult

8.342.1 Detailed Description

This structure contains information of the request parameters associated with a Unblock PIN API.

Parameters

EncryptedPIN1	• See uim_encryptedPIN1 for more information.
sessionInfo	• See uim_sessionInformation for more information.
pinProtection	• See uim_unblockUIMPIN for more information.

pKeyReferenceI-D(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

8.342.2 Field Documentation

8.342.2.1 uim_encryptedPIN1 pack_uim_UnblockPin_t::EncryptedPIN1

8.342.2.2 uint32_t* pack_uim_UnblockPin_t::pIndicationToken

8.342.2.3 uim_unblockUIMPIN pack_uim_UnblockPin_t::pinProtection

8.342.2.4 uint8_t* pack_uim_UnblockPin_t::pKeyReferenceID

8.342.2.5 uim_sessionInformation pack_uim_UnblockPin_t::sessionInfo

8.342.2.6 uint16_t pack_uim_UnblockPin_t::Tlvresult

8.343 pack_uim_VerifyPin_t Struct Reference

Data Fields

- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint8_t * pKeyReferenceID
- uim_sessionInformation sessionInfo
- uim_verifyUIMPIN verifyPIN
- uint16_t Tlvresult

8.343.1 Detailed Description

This structure contains information of the request parameters associated with a verify PIN API.

Parameters

sessionInfo	• See uim_sessionInformation for more information.
verifyPIN	• See uim_verifyUIMPIN for more information.

pEncryptedPIN1(optional)	• See uim_encryptedPIN1 for more information.
pKeyReferenceID(optional)	• Indicates the PIN key reference ID. • Indicates the PIN key reference ID. Valid values are from 1 to 8, respectively, for application 1 to application 8. • This TLV is used only for PIN1 and PIN2 and is ignored in all other cases.
pIndicationToken(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication. • Valid Values – 0 - Result of operation in response. Indication will not be generated by the modem – Any other positive number - Result of operation in indication. Indication will have same token value set by this function

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.343.2 Field Documentation

8.343.2.1 uim_encryptedPIN1∗ pack_uim_VerifyPin_t::pEncryptedPIN1

8.343.2.2 uint32_t∗ pack_uim_VerifyPin_t::pIndicationToken

8.343.2.3 uint8_t∗ pack_uim_VerifyPin_t::pKeyReferenceID

8.343.2.4 uim_sessionInformation pack_uim_VerifyPin_t::sessionInfo

8.343.2.5 uint16_t pack_uim_VerifyPin_t::Tlvresult

8.343.2.6 uim_verifyUIMPIN pack_uim_VerifyPin_t::verifyPIN

8.344 pack_voice_AnswerUSSD_t Struct Reference

Data Fields

- uint8_t ∗ pInfo

8.344.1 Detailed Description

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.344.2 Field Documentation

8.344.2.1 uint8_t* pack_voice_AnswerUSSD_t::pInfo

8.345 pack_voice_OriginateUSSD_t Struct Reference

Data Fields

- uint8_t * pInfo

8.345.1 Detailed Description

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.345.2 Field Documentation

8.345.2.1 uint8_t* pack_voice_OriginateUSSD_t::pInfo

8.346 pack_voice_SLQSOriginateUSSD_t Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.346.1 Detailed Description

This structure contains USS Information

Parameters

ussDCS	• 1 - ASCII coding scheme • 2 - 8-BIT coding scheme • 3 - UCS2
ussLen	• Range 1 to 182
ussData	• Data encoded as per the DCS

8.346.2 Field Documentation

8.346.2.1 uint8_t pack_voice_SLQSOriginateUSSD_t::ussData[182]

8.346.2.2 uint8_t pack_voice_SLQSOriginateUSSD_t::ussDCS

8.346.2.3 uint8_t pack_voice_SLQSOriginateUSSD_t::ussLen

8.347 pack_voice_SLQSVoiceALSSelectLine_t Struct Reference

Data Fields

- uint8_t lineValue

8.347.1 Detailed Description

This structure contains ALS Select Line Information Parameters.

Parameters

lineValue	• ALS Line Value. – 0x00 - ALS_LINE1 - Line 1 (default) – 0x01 - ALS_LINE2 - Line 2

8.347.2 Field Documentation

8.347.2.1 uint8_t pack_voice_SLQSVoiceALSSelectLine_t::lineValue

8.348 pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference

Data Fields

- uint8_t switchOption

8.348.1 Detailed Description

This structure contains ALS Set Line Switching Information Parameters.

Parameters

switchOption	• Switch Option. – 0x00 - VOICE_LINE_SWITCHING_NOT_ALLOWED - Line switching is not allowed – 0x01 - VOICE_LINE_SWITCHING_ALLOWED - Line switching is allowed

8.348.2 Field Documentation

8.348.2.1 uint8_t pack_voice_SLQSVoiceALSSetLineSwitching_t::switchOption

8.349 pack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.349.1 Detailed Description

Contains the parameters passed for pack voice Answer Call.

Parameters

pCallId	• Unique call identifier for the call that must be answered.

8.349.2 Field Documentation

8.349.2.1 uint8_t* pack_voice_SLQSVoiceAnswerCall_t::pCallId

8.350 pack_voice_SLQSVoiceBindSubscription_t Struct Reference

Data Fields

- uint8_t subsType

8.350.1 Detailed Description

This structure contains Bind Subscription Information Parameters.

Parameters

subsType	• Subscription Type. – 0x00 - VOICE_SUBS_TYPE_PRIMARY - Primary – 0x01 - VOICE_SUBS_TYPE_SECONDARY - Secondary

8.350.2 Field Documentation

8.350.2.1 uint8_t pack_voice_SLQSVoiceBindSubscription_t::subsType

8.351 pack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- voice_burstDTMFInfo BurstDTMFInfo
- voice_DTMFLengths * pBurstDTMFLengths

8.351.1 Detailed Description

This structure contains parameters of burst Dual-Tone Multifrequency (DTMF)

Parameters

BurstDTMFInfo	• Burst DTMF Information – See voice_burstDTMFInfo for more information
pBurstDTMF-Lengths	[optional] • DTMF Lengths – See voice_DTMFLengths for more information

8.351.2 Field Documentation

8.351.2.1 voice_burstDTMFInfo pack_voice_SLQSVoiceBurstDTMF_t::BurstDTMFInfo

8.351.2.2 voice_DTMFLengths* pack_voice_SLQSVoiceBurstDTMF_t::pBurstDTMFLengths

8.352 pack_voice_SLQSVoiceDialCall_t Struct Reference

Data Fields

- uint8_t callNumber [81]
- uint8_t * pCallType
- uint8_t * pCLIRType
- voice_UUSInfo * pUUSInFo
- voice_CUGInfo * pCUGInfo
- uint8_t * pEmergencyCategory
- voice_calledPartySubAdd * pCallPartySubAdd
- uint8_t * pSvcType

8.352.1 Detailed Description

This structure contains Voice Call Request Parameters

Parameters

callNumber[81]	• Number to be dialed in ASCII string, NULL terminated. • Length Range [1 to 81]
pCall-Type(optional)	• the type of call to be dialed. CALL_TYPE_VOICE is automatically selected if this parameter is not provided. When CALL_TYPE_NON_STD_OTASP is selected, the call is sent as a nonstandard OTASP call regardless of the digit string Call type values are: – 0x00 - CALL_TYPE_VOICE - Voice (automatic selection) – 0x01 - CALL_TYPE_VOICE_FORCED - Avoid modem call classification – 0x08 - CALL_TYPE_NON_STD_OTASP - Nonstandard OTASP* – 0x09 - CALL_TYPE_EMERGENCY - Emergency
pCLIR-Type(optional)	• CLIR type values are: – 0x01 - CLIR_SUPPRESSION - Suppression – 0x02 - CLIR_INVOCATION - Invocation

pUUSInFo(optional)	• Pointer to structure of UUSInfo – See voice_UUSInfo for more information
pCUGInfo(optional)	• Pointer to structure of CUGInfo – See voice_CUGInfo for more information
pEmergencyCategory(optional)	• Bit mask of emergency number categories. This is only applicable when the call type is set to Emergency. – Bit 0 - VOICE_EMER_CAT_POLICE_BIT - Police – Bit 1 - VOICE_EMER_CAT_AMBULANCE_BIT - Ambulance – Bit 2 - VOICE_EMER_CAT_FIRE_BRIGADE_BIT- Fire brigade – Bit 3 - VOICE_EMER_CAT_MARINE_GUARD_BIT - Marine guard – Bit 4 - VOICE_EMER_CAT_MOUNTAIN_RESCUE_BIT - Mountain rescue – Bit 5 - VOICE_EMER_CAT_MANUAL_ECALL_BIT - Manual emergency call – Bit 6 - VOICE_EMER_CAT_AUTO_ECALL_BIT - Automatic emergency call – Bit 7 - VOICE_EMER_CAT_SPARE_BIT - Spare bit
pCallPartySubAdd(optional)	• Pointer to structure of calledPartySubAdd – See voice_calledPartySubAdd for more information
pSvcType(optional)	• Service Type. – 0x01 - VOICE_DIAL_CALL_SRV_TYPE_AUTOMATIC - Automatic – 0x02 - VOICE_DIAL_CALL_SRV_TYPE_GSM - GSM – 0x03 - VOICE_DIAL_CALL_SRV_TYPE_WCDMA - WCDMA – 0x04 - VOICE_DIAL_CALL_SRV_TYPE_CDMA_AUTOMATIC - CDMA automatic – 0x05 - VOICE_DIAL_CALL_SRV_TYPE_GSM_WCDMA - GSM or WCDMA – 0x06 - VOICE_DIAL_CALL_SRV_TYPE_LTE -LTE

8.352.2 Field Documentation

8.352.2.1 uint8_t pack_voice_SLQSVoiceDialCall_t::callNumber[81]

8.352.2.2 voice_calledPartySubAdd* pack_voice_SLQSVoiceDialCall_t::pCallPartySubAdd

8.352.2.3 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCallType

8.352.2.4 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCLIRType

8.352.2.5 voice_CUGInfo* pack_voice_SLQSVoiceDialCall_t::pCUGInfo

8.352.2.6 uint8_t* pack_voice_SLQSVoiceDialCall_t::pEmergencyCategory

8.352.2.7 uint8_t* pack_voice_SLQSVoiceDialCall_t::pSvcType

8.352.2.8 voice_UUSInfo* pack_voice_SLQSVoiceDialCall_t::pUUSInFo

8.353 pack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.353.1 Detailed Description

Parameters

pCallId	• Unique call identifier for the call that must be ended

8.353.2 Field Documentation

8.353.2.1 uint8_t* pack_voice_SLQSVoiceEndCall_t::pCallId

8.354 pack_voice_SLQSVoiceGetCallBarring_t Struct Reference

Data Fields

- uint8_t reason
- uint8_t * pSvcClass

8.354.1 Detailed Description

This structure contains Voice Get Call Barring Pack Parameters

Parameters

reason	• Call Barring Reason • Values: – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME - Outgoing external to home – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available

8.354.2 Field Documentation

8.354.2.1 uint8_t* pack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.354.2.2 uint8_t pack_voice_SLQSVoiceGetCallBarring_t::reason

8.355 pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t * pSvcClass

8.355.1 Detailed Description

This structure contains Voice Get Call Forwarding Status Pack Parameters

Parameters

Reason	• Call Forwarding Reason • Values: – 0x01 - QMI_VOICE_REASON_FWDREASON_UNCONDITIONAL - Unconditional call forwarding – 0x02 - QMI_VOICE_REASON_FWDREASON_MOBILEBUSY - Forward when the mobile is busy – 0x03 - QMI_VOICE_REASON_FWDREASON_NOREPLY - Forward when there is no reply – 0x04 - QMI_VOICE_REASON_FWDREASON_UNREACHABLE - Forward when the call is unreachable – 0x05 - QMI_VOICE_REASON_FWDREASON_ALLFORWARDING - All forwarding – 0x06 - QMI_VOICE_REASON_FWDREASON_ALLCONDITIONAL - All conditional forwarding
pSvcClass(optional)	• Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.

8.355.2 Field Documentation

8.355.2.1 uint8_t* pack_voice_SLQSVoiceGetCallForwardingStatus_t::pSvcClass

8.355.2.2 uint8_t pack_voice_SLQSVoiceGetCallForwardingStatus_t::Reason

8.356 pack_voice_SLQSVoiceGetCallInfo_t Struct Reference

Data Fields

- uint8_t callID

8.356.1 Detailed Description

This structure contains information of the request parameters associated with a call.

Parameters

callID	• Call identifier for the call queried for information.

8.356.2 Field Documentation

8.356.2.1 uint8_t pack_voice_SLQSVoiceGetCallInfo_t::callID

8.357 pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference

Data Fields

- uint8_t * pSvcClass

8.357.1 Detailed Description

This structure contains Voice Get Call Waiting Response Parameters

Parameters

pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available

8.357.2 Field Documentation

8.357.2.1 uint8_t* pack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.358 pack_voice_SLQSVoiceGetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnswer
- uint8_t * pAirTimer
- uint8_t * pRoamTimer
- uint8_t * pTTYMode
- uint8_t * pPrefVoiceSO
- uint8_t * pAMRStatus
- uint8_t * pPrefVoicePrivacy
- uint8_t * pNamID
- uint8_t * pVoiceDomainPref

8.358.1 Detailed Description

This structure contains Voice Get Configuration Request Parameters

Parameters

pAuto-Answer(optional)	• Indicator to retrieve the Auto Answer Information. – 0x01 - Include auto answer information
pAir-Timer(optional)	• Indicator to retrieve the Air Timer Information. – 0x01 - Include air calls timer count information • Currently Not Supported.
pRoam-Timer(optional)	• Indicator to retrieve the Roam Timer Information. – 0x01 - Include roam calls timer information • Currently Not Supported.
pTTY-Mode(optional)	• Indicator to retrieve the TTY Mode Information. – 0x01 - Include TTY configuration status information
pPrefVoiceS-O(optional)	• Indicator to retrieve the Preferred Voice SO Information. – 0x01 - Include preferred voice configuration status information • Currently Not Supported.
pAMR-Status(optional)	• Indicator to retrieve the AMR Status Information. – 0x01 - Include AMR status information
pPrefVoice-Privacy(optional)	• Indicator to retrieve the Preferred Voice Privacy Information. – 0x01 - Include preferred voice privacy status information
pNamI-D(optional)	• Index of the Number Assignment Module Index (CDMA subscription) to be configured • Range: 0 to 3. • Some modems support only 1 or 2 NAMs. • The NAM Index is valid only when the request contains at least one of Air Timer, Roam Timer, and Preferred Voice SO. • If no nam_id value is specified in the request,the default value is 0.
pVoiceDomain-Pref(optional)	• Indicator to retrieve the Preferred Voice Domain Information. – 0x01 - Include voice domain preference information

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.358.2 Field Documentation

8.358.2.1 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAirTimer

8.358.2.2 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAMRStatus

8.358.2.3 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAutoAnswer

8.358.2.4 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pNamID

8.358.2.5 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoicePrivacy

8.358.2.6 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoiceSO

8.358.2.7 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pRoamTimer

8.358.2.8 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pTTYMode

8.358.2.9 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pVoiceDomainPref

8.359 pack_voice_SLQSVoiceIndicationRegister_t Struct Reference

Data Fields

- uint8_t * pRegDTMFEvents
- uint8_t * pRegVoicePrivacyEvents
- uint8_t * pSuppsNotifEvents

8.359.1 Detailed Description

This structure contains parameters of Indication Register Information

Parameters

pRegDTMF-Events(optional)	• Registration Indication For DTMF Events. • When this registration is enabled, the device learns of DTMF events via the QMI_VOICE_DTMF_IND indication. – 0x00 - Disable – 0x01 - Enable
pRegVoice-Privacy-Events(optional)	• Registration Indication For Voice Privacy Events. • When this registration is enabled, the device learns of DTMF events via the QMI_VOICE_PRIVACY_IND indication. – 0x00 - Disable – 0x01 - Enable
pSuppsNotif-Events(optional)	• Registration Indication For Supplementary Service Notification Events. • When this registration is enabled, the device learns of DTMF events via the QMI_VOICE_SUPS_NOTIFICATION_IND indication. – 0x00 - Disable – 0x01 - Enable

Note

One of the optional parameter is mandatory to be present in the request.

8.359.2 Field Documentation

8.359.2.1 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegDTMFEvents

8.359.2.2 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegVoicePrivacyEvents

8.359.2.3 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pSuppsNotifEvents

8.360 pack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint8_t SUPSType
- uint8_t * pCallID

8.360.1 Detailed Description

This structure contains Manage Calls Information.

Parameters

SUPSType	• Supplementary service type during the call. – 0x01 - SUPS_TYPE_RELEASE_HELD_OR_WAITING * Release is held or waiting – 0x02 - SUPS_TYPE_RELEASE_ACTIVE_ACCEPT_HELD_OR_WAITING * Release is active and accepting held or waiting – 0x03 - SUPS_TYPE_HOLD_ACTIVE_ACCEPT_WAITING_OR_HELD * Hold is active and accepting waiting or held – 0x04 - SUPS_TYPE_HOLD_ALL_EXCEPT_SPECIFIED_CALL * Hold all calls except a specified one – 0x05 - SUPS_TYPE_MAKE_CONFERENCE_CALL * Make a conference call – 0x06 - SUPS_TYPE_EXPLICIT_CALL_TRANSFER * Explicit call transfer – 0x07 - SUPS_TYPE_CCBS_ACTIVATION * Activate completion of calls to busy subscriber – 0x08 - SUPS_TYPE_END_ALL_CALLS * End all calls – 0x09 - SUPS_TYPE_RELEASE_SPECIFIED_CALL * Release a specified call
pCallID[Optional]	• Applicable only for SUPSType 0x04, 0x07, and 0x09

8.360.2 Field Documentation

8.360.2.1 uint8_t* pack_voice_SLQSVoiceManageCalls_t::pCallID

8.360.2.2 uint8_t pack_voice_SLQSVoiceManageCalls_t::SUPSType

8.361 pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference

Data Fields

- struct voice_USSInfo USSInformation

8.361.1 Detailed Description

This structure contains Orig USSD No Wait Information Parameters.

Parameters

USSInformation	• See voice_USSInfo for more information.

8.361.2 Field Documentation

8.361.2.1 struct voice_USSInfo pack_voice_SLQSVoiceOrigUSSDNoWait_t::USSInformation

8.362 pack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t * pFlashPayLd
- uint8_t * pFlashType

8.362.1 Detailed Description

This structure contains the flash information associated with a call.

Parameters

pCallID	• Unique call identifier associated with the current call.
pFlashPayLd(optional)	• Payload in ASCII to be sent in Flash. • Variable Length, NULL terminated.
pFlashType(optional)	• Flash type. – 0 - Simple Flash (default) – 1 - Activate answer hold – 2 - Deactivate answer hold

8.362.2 Field Documentation

8.362.2.1 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pCallID

8.362.2.2 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashPayLd

8.362.2.3 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashType

8.363 pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t oldPasswd [4]
- uint8_t newPasswd [4]
- uint8_t newPasswdAgain [4]

8.363.1 Detailed Description

This structure contains Voice Set Call Barring Password Pack Parameters

Parameters

Reason	• Call Barring Reason • Values: – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME - Outgoing external to home – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
oldPasswd[BARRING_PASSWORD_LENGTH]	• Old password. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.
newPasswd[BARRING_PASSWORD_LENGTH]	• New password. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.
newPasswdAgain[BARRING_PASSWORD_LENGTH]	• New password Again. – Password consists of 4 ASCII digits. – Range: 0000 to 9999.

8.363.2 Field Documentation

8.363.2.1 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswd[4]

8.363.2.2 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswdAgain[4]

8.363.2.3 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::oldPasswd[4]

8.363.2.4 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::Reason

8.364 pack_voice_SLQSVoiceSetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnswer
- voice_airTimer * pAirTimerConfig
- voice_roamTimer * pRoamTimerConfig
- uint8_t * pTTYMode
- voice_prefVoiceSO * pPrefVoiceSO
- uint8_t * pPrefVoiceDomain

8.364.1 Detailed Description

This structure contains information about the Set Configuration Request Parameters.

Parameters

pAutoAnswer	• Value specified is written to NV_AUTO_ANSWER_I. (optional) • Values: – 0x00 - Disable – 0x01 - Enable
pAirTimerConfig	• Value specified is written to NV_AIR_CNT_I. (optional) • See voice_airTimer for more information
pRoamTimer-Config	• Value specified is written to NV_ROAM_CNT_I. (optional) • See voice_roamTimer for more information
pTTYMode	• Value specified is written to NV_TTY_I. (optional) • Values: – 0x00 - TTY_MODE_FULL - Full – 0x01 - TTY_MODE_VCO - Voice carry over – 0x02 - TTY_MODE_HCO - Hearing carry over – 0x03 - TTY_MODE_OFF - Off
pPrefVoiceSO	• Value specified is written to NV_PREF_VOICE_SO_I. (optional) • See voice_prefVoiceSO for more information

pPrefVoice-Domain	• Preferred Voice-Domain. (optional) • Values: – 0x00 - VOICE_DOMAIN_PREF_CS_ONLY - Circuit-switched (CS) only – 0x01 - VOICE_DOMAIN_PREF_PS_ONLY - Packet-switched (PS) only – 0x02 - VOICE_DOMAIN_PREF_CS_PREF - CS is preferred, PS is secondary – 0x03 - VOICE_DOMAIN_PREF_PS_PREF - PS is preferred, CS is secondary

Note

One of the optional parameters must be present in the request.

8.364.2 Field Documentation

8.364.2.1 voice_airTimer* pack_voice_SLQSVoiceSetConfig_t::pAirTimerConfig

8.364.2.2 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pAutoAnswer

8.364.2.3 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceDomain

8.364.2.4 voice_prefVoiceSO* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSO

8.364.2.5 voice_roamTimer* pack_voice_SLQSVoiceSetConfig_t::pRoamTimerConfig

8.364.2.6 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pTTYMode

8.365 pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference

Data Fields

- uint8_t privacyPref

8.365.1 Detailed Description

This structure contains the preferred voice privacy values.

Parameters

privacyPref	• Voice Privacy Preference – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy

8.365.2 Field Documentation

8.365.2.1 uint8_t pack_voice_SLQSVoiceSetPreferredPrivacy_t::privacyPref

8.366 pack_voice_SLQSVoiceSetSUPSService_t Struct Reference

Data Fields

- uint8_t voiceSvc
- uint8_t reason
- uint8_t * pServiceClass
- uint8_t * pCallBarringPasswd
- uint8_t * pCallForwardingNumber
- uint8_t * pTimerVal
- voice_callFwdTypeAndPlan * pCallFwdTypeAndPlan

8.366.1 Detailed Description

This structure contains Supplementary Service request parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

voiceSvc	• Manages all call-independent supplementary services, such as activation, deactivation, registration, and erasure (mandatory) – 0x01 - VOICE_SERVICE_ACTIVATE – 0x02 - VOICE_SERVICE_DEACTIVATE – 0x03 - VOICE_SERVICE_REGISTER – 0x04 - VOICE_SERVICE_ERASE

reason	• supplementary service reason values (mandatory) – 0x01 - QMI_VOICE_REASON_FWD_UNCONDITIONAL Unconditional call forwarding – 0x02 - QMI_VOICE_REASON_FWD_MOBILEBUSY Forward when the mobile is busy – 0x03 - QMI_VOICE_REASON_FWD_NOREPLY Forward when there is no reply – 0x04 - QMI_VOICE_REASON_FWD_UNREACHABLE Forward when the call is unreachable – 0x05 - QMI_VOICE_REASON_FWD_ALLFORWARDING All forwarding – 0x06 - QMI_VOICE_REASON_FWD_ALLCONDITIONAL All conditional forwarding – 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING All outgoing calls are barred – 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT Outgoing internal calls are barred – 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME Outgoing calls external to home are barred – 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING All incoming calls are barred – 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING Roaming incoming calls are barred – 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING All calls are barred – 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING All outgoing calls are barred – 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING All incoming calls are barred – 0x0F - QMI_VOICE_REASON_CALLWAITING Call waiting
pServiceClass	• Service class is a combination (sum) of information class constants (optional) – See liteServiceClassInformation for more information
pCallBarring-Passwd	• Password is required if call barring is provisioned using a password. Password consists of 4 ASCII digits. Range: 0000 to 9999 (optional)
pCallForwarding-Number	• Call forwarding number to be registered with the network. This has to be included in the request only when the service is set to VOICE_SERVICE_REGISTER. NULL terminated ASCII string. (optional)
pTimerVal	• Call forwarding no reply timer value in seconds. This has to be included in the request only when the service is set to VOICE_SERVICE_REGISTER and the reason is QMI_VOICE_REASON_FWD_NOREPLY. (optional) – Range: 5 to 30 in steps of 5

pCallFwdType-AndPlan	• Information about call forwarding type and plan. This parameter is ignored when the Call Forwarding Number is not included (optional) – See voice_callFwdTypeAndPlan for more information

8.366.2 Field Documentation

8.366.2.1 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallBarringPasswd

8.366.2.2 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallForwardingNumber

8.366.2.3 voice_callFwdTypeAndPlan* pack_voice_SLQSVoiceSetSUPSService_t::pCallFwdTypeAndPlan

8.366.2.4 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pServiceClass

8.366.2.5 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pTimerVal

8.366.2.6 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::reason

8.366.2.7 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::voiceSvc

8.367 pack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t DTMFdigit

8.367.1 Detailed Description

This structure contains parameters of continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Start continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device
DTMFdigit	• DTMF digit in ASCII.

8.367.2 Field Documentation

8.367.2.1 uint8_t pack_voice_SLQSVoiceStartContDTMF_t::DTMFdigit

8.367.2.2 uint8_t* pack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.368 pack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID

8.368.1 Detailed Description

This structure contains parameters of stop continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Stop continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device

8.368.2 Field Documentation

8.368.2.1 uint8_t pack_voice_SLQSVoiceStopContDTMF_t::callID

8.369 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference

Data Fields

- uint8_t * pEnableNotification

8.369.1 Detailed Description

WDS SWI DHCPv4 Client Lease Change Structure

Parameters

pEnable-Notification	• Enable Notification or not

8.369.2 Field Documentation

8.369.2.1 uint8_t* pack_wds_DHCPv4ClientLeaseChange_t::pEnableNotification

8.370 pack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t profiletype

8.370.1 Detailed Description

Parameters

profiletype	profile type

8.370.2 Field Documentation

8.370.2.1 uint32_t pack_wds_GetDefaultProfile_t::profiletype

8.371 pack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family

8.371.1 Detailed Description

Parameters

type	profile type • 0 - 3GPP • 1 - 3GPP2
type	profile family • 0 - Embedded • 1 - Tethered

8.371.2 Field Documentation

8.371.2.1 uint8_t pack_wds_GetDefaultProfileNum_t::family

8.371.2.2 uint8_t pack_wds_GetDefaultProfileNum_t::type

8.372 pack_wds_GetDormancyState_t Struct Reference

8.372.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.373 pack_wds_GetLastMobileIPError_t Struct Reference

8.373.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.374 pack_wds_GetMobileIP_t Struct Reference

8.374.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.375 pack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t index

8.375.1 Detailed Description

Parameters

index	• Mobile IP profile ID

8.375.2 Field Documentation

8.375.2.1 uint8_t pack_wds_GetMobileIPProfile_t::index

8.376 pack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * pStatMask

8.376.1 Detailed Description

Parameters

pStatMask	• Packet Statistics Mask – 0x00000001 - Tx packets OK – 0x00000002 - Rx packets OK – 0x00000004 - Tx packet errors – 0x00000008 - Rx packet errors – 0x00000010 - Tx overflows – 0x00000020 - Rx overflows – 0x00000040 - Tx bytes OK – 0x00000080 - Rx bytes OK

8.376.2 Field Documentation

8.376.2.1 uint32_t* pack_wds_GetPacketStatistics_t::pStatMask

8.377 pack_wds_GetPacketStatus_t Struct Reference

Data Fields

- uint32_t statmask

8.377.1 Detailed Description

Parameters

statmask	packet statistics mask

8.377.2 Field Documentation

8.377.2.1 uint32_t pack_wds_GetPacketStatus_t::statmask

8.378 pack_wds_GetSessionDuration_t Struct Reference

8.378.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.379 pack_wds_RMSetTransferStatistics_t Struct Reference

Data Fields

- rmTrasnferStaticsReq RmTrasnferStaticsReq

8.379.1 Detailed Description

Parameters

RmTrasnfer-StaticsReq[IN]	• See rmTrasnferStaticsReq for more information

8.379.2 Field Documentation

8.379.2.1 rmTrasnferStaticsReq pack_wds_RMSetTransferStatistics_t::RmTrasnferStaticsReq

8.380 pack_wds_SetAutoconnect_t Struct Reference

Data Fields

- uint8_t acsetting
- uint8_t acroamsetting

8.380.1 Detailed Description

auto connect data session parameters.

Parameters

acsetting	• Current autoconnect setting: – 0x00 - Autoconnect disabled – 0x01 - Autoconnect enabled – 0x02 - Autoconnect paused (resume on powercycle)
acroamsetting	• Current autoconnect roaming status – 0x00 - Autoconnect always allowed – 0x01 - Autoconnect while in home service area only

8.380.2 Field Documentation

8.380.2.1 uint8_t pack_wds_SetAutoconnect_t::acroamsetting

8.380.2.2 uint8_t pack_wds_SetAutoconnect_t::acsetting

8.381 pack_wds_SetDefaultProfile_t Struct Reference

Data Fields

- uint32_t profileType
- uint32_t pdpType
- uint32_t ipAddress
- uint32_t primaryDNS
- uint32_t secondaryDNS
- uint32_t authentication

- uint8_t * pName
- uint8_t * pUsername
- uint8_t * pApnname
- uint8_t * pPassword

8.381.1 Detailed Description

Writes the default profile settings to the device. The default profile is used to establish an autoconnect data session.

Parameters

profileType	• Type of profile – 0 - 3GPP – 1 - 3GPP2
pdpType	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile (optional) – 0 - PDP-IP (IPv4) – 1 - PDP-PPP – 2 - PDP-IPv6 – 3 - PDP-IPv4v6
ipAddress	• Preferred IPv4 addr to be assigned to device (optional)
primaryDNS	• Primary DNS Ipv4 address preference (optional)
secondaryDNS	• Secondary DNS Ipv4 address preference (optional)
authentication	• Bitmap that indicates authentication algorithm preference (optional) – 0x00000001 - PAP preference * 0 - Never performed * 1 - May be performed – 0x00000002 - CHAP preference * 0 - Never performed * 1 - May be performed – All other bits are reserved and must be set to 0 – If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.
pName	• profile Name (optional)
pAPNName	• Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network (optional) • If value is NULL or omitted, then subscription default value will be requested.

pUsername	• Username used during network authentication (optional)
pPassword	• Password used during network authentication (optional)

8.381.2 Field Documentation

8.381.2.1 uint32_t pack_wds_SetDefaultProfile_t::authentication

8.381.2.2 uint32_t pack_wds_SetDefaultProfile_t::ipAddress

8.381.2.3 uint8_t* pack_wds_SetDefaultProfile_t::pApnname

8.381.2.4 uint32_t pack_wds_SetDefaultProfile_t::pdpType

8.381.2.5 uint8_t* pack_wds_SetDefaultProfile_t::pName

8.381.2.6 uint8_t* pack_wds_SetDefaultProfile_t::pPassword

8.381.2.7 uint32_t pack_wds_SetDefaultProfile_t::primaryDNS

8.381.2.8 uint32_t pack_wds_SetDefaultProfile_t::profileType

8.381.2.9 uint8_t* pack_wds_SetDefaultProfile_t::pUsername

8.381.2.10 uint32_t pack_wds_SetDefaultProfile_t::secondaryDNS

8.382 pack_wds_SetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family
- uint8_t index

8.382.1 Detailed Description

This structure to hold Set default profile number

Parameters

type	• 0 - 3GPP • 1 - 3GPP2
family	• 0 - Embedded • 1 - Tethered
index	

8.382.2 Field Documentation

8.382.2.1 uint8_t pack_wds_SetDefaultProfileNum_t::family

8.382.2.2 uint8_t pack_wds_SetDefaultProfileNum_t::index

8.382.2.3 uint8_t pack_wds_SetDefaultProfileNum_t::type

8.383 pack_wds_SetMobileIP_t Struct Reference

Data Fields

- uint32_t mode

8.383.1 Detailed Description

Parameters

mode	• Mobile IP setting – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only

8.383.2 Field Documentation

8.383.2.1 uint32_t pack_wds_SetMobileIP_t::mode

8.384 pack_wds_SetMobileIPParameters_t Struct Reference

Data Fields

- char * pSPC
- uint32_t * pMode
- uint8_t * pRetryLimit
- uint8_t * pRetryInterval
- uint8_t * pReRegPeriod
- uint8_t * pReRegTraffic
- uint8_t * pHAAuthenticator
- uint8_t * pHA2002bis

8.384.1 Detailed Description

Mobile IP parameters information.

Parameters

pSPC	• NULL-terminated string representing six digit service programming code.

pMode	• Mode to be set (optional) – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only
pRetryLimit	• Registration retry attempt limit (optional)
pRetryInterval	• Registration retry attempt interval used to determine the time between registration attempts (optional)
pReRegPeriod	• Period (in minutes) to attempt re-registration before current registration expires (optional)
pReRegTraffic	• Re-registration only if traffic since last attempt (optional) – Zero - Disabled – NonZero - Enabled
pHA-Authenticator	• MH-HA authenticator calculator (optional) – Zero - Disabled – NonZero - Enabled
pHA2002bis	• MH-HA RFC 2002bis authentication instead of RFC2002 (optional) – Zero - Disabled – NonZero - Enabled

8.384.2 Field Documentation

8.384.2.1 uint8_t* pack_wds_SetMobileIPParameters_t::pHA2002bis

8.384.2.2 uint8_t* pack_wds_SetMobileIPParameters_t::pHAAuthenticator

8.384.2.3 uint32_t* pack_wds_SetMobileIPParameters_t::pMode

8.384.2.4 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegPeriod

8.384.2.5 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegTraffic

8.384.2.6 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryInterval

8.384.2.7 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryLimit

8.384.2.8 char* pack_wds_SetMobileIPParameters_t::pSPC

8.385 pack_wds_SetMobileIPProfile_t Struct Reference

Data Fields

- int8_t spc [10]
- uint8_t index
- uint8_t * pEnabled
- uint32_t * pAddress
- uint32_t * pPrimaryHA
- uint32_t * pSecondaryHA
- uint8_t * pRevTunneling
- int8_t * pNAI
- uint32_t * pHASPI
- uint32_t * pAAASPI
- int8_t * pMNHA
- int8_t * pMNAAA

8.385.1 Detailed Description

Sets the mobile IP parameters.

Parameters

spc	• Six digit service programming code string
index	• Index of the profile to modify
pEnabled	• (Optional) Enable profile? 0 - Disabled Nonzero - Enabled
pAddress	• (Optional) Home IPv4 address
pPrimaryHA	• (Optional) Primary home agent IPv4 address
pSecondaryHA	• (Optional) Secondary home agent IPv4 address
pRevTunneling	• (Optional) Enable reverse tunneling? 0 - Disabled Nonzero - Enabled
pNAI	• (Optional) Network access identifier string
pHASPI	• (Optional) Home agent security parameter index
pAAASPI	• (Optional) AAA server security parameter index
pMNHA	• (Optional) MN-HA key string

pMNAAA	• (Optional) MN-AAA key string

8.385.2 Field Documentation

8.385.2.1 uint8_t pack_wds_SetMobileIPProfile_t::index

8.385.2.2 uint32_t* pack_wds_SetMobileIPProfile_t::pAAASPI

8.385.2.3 uint32_t* pack_wds_SetMobileIPProfile_t::pAddress

8.385.2.4 uint8_t* pack_wds_SetMobileIPProfile_t::pEnabled

8.385.2.5 uint32_t* pack_wds_SetMobileIPProfile_t::pHASPI

8.385.2.6 int8_t* pack_wds_SetMobileIPProfile_t::pMNAAA

8.385.2.7 int8_t* pack_wds_SetMobileIPProfile_t::pMNHA

8.385.2.8 int8_t* pack_wds_SetMobileIPProfile_t::pNAI

8.385.2.9 uint32_t* pack_wds_SetMobileIPProfile_t::pPrimaryHA

8.385.2.10 uint8_t* pack_wds_SetMobileIPProfile_t::pRevTunneling

8.385.2.11 uint32_t* pack_wds_SetMobileIPProfile_t::pSecondaryHA

8.385.2.12 int8_t pack_wds_SetMobileIPProfile_t::spc[10]

8.386 pack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo * pCurProfile

8.386.1 Detailed Description

Parameters

ProfileID	• 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile
ProfileType	• Identifies the technology type of the profile – 0x00 - 3GPP – 0x01 - 3GPP2 – NULL is not allowed

curProfile	• union of 3GPP and 3GPP2 profile

Note

- If profileID is NULL, 3GPP profile will be created and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be created

8.386.2 Field Documentation

8.386.2.1 wds_profileInfo* pack_wds_SLQSCreateProfile_t::pCurProfile

8.386.2.2 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileId

8.386.2.3 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileType

8.387 pack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.387.1 Detailed Description

This structure contains the information about the profile to be deleted.

Parameters

profileType	• Identifies the type of profile – 0x00 - 3GPP • Note: Deletion of 3GPP2 profiles is not supported.
profileIndex	• Index of the configured profile to be deleted – Value between 1 - 16 (EM/MC73xx or earlier) – Value between 1 - 24 (EM/MC74xx onwards)

8.387.2 Field Documentation

8.387.2.1 uint8_t pack_wds_SLQSDeleteProfile_t::profileIndex

8.387.2.2 uint8_t pack_wds_SLQSDeleteProfile_t::profileType

8.388 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

8.388.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.389 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference

8.389.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.390 pack_wds_SLQSGetDUNCallInfo_t Struct Reference

Data Fields

- uint32_t Mask
- uint8_t * pReportConnStatus
- transferStatInd * pTransferStatInd
- uint8_t * pReportDormStatus
- uint8_t * pReportDataBearerTech
- uint8_t * pReportChannelRate

8.390.1 Detailed Description

This structure contains the DUN Call Info Request parameters.

Parameters

Mask	• Mandatory parameter • Set the bits corresponding to the information requested to 1 • All other bits must be set to 0. • If any values are not available or applicable, the corresponding TLVs are not returned in the response. – Bit 0 - Connection Status – Bit 1 - Last call end reason – Bit 2 - Tx/Rx bytes OK – Bit 3 - Dormancy status – Bit 4 - Data bearer – Bit 5 - Channel rate – Bit 6 - Call active duration
pReportConn-Status	• Connect Status Indicator – 0 - Do not report – 1 - Report connection status and call end reason

pTransferStatInd	• See TransferStatInd for more information
pReportDorm-Status	• Dormancy Status Indicator – 0 - Do not report – 1 - Report traffic channel state of interface used for data connection
pReportData-BearerTech	• Current Data Bearer Technology Indicator – 0 - Do not report – 1 - Report radio interface used for data transfer when it changes
pReport-ChannelRate	• Channel Rate Indicator – 0 - Do not report – 1 - Report channel rate

8.390.2 Field Documentation

8.390.2.1 uint32_t pack_wds_SLQSGetDUNCallInfo_t::Mask

8.390.2.2 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportChannelRate

8.390.2.3 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportConnStatus

8.390.2.4 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDataBearerTech

8.390.2.5 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDormStatus

8.390.2.6 transferStatInd* pack_wds_SLQSGetDUNCallInfo_t::pTransferStatInd

8.391 pack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- uint8_t ProfileId
- uint8_t ProfileType

8.391.1 Detailed Description

Parameters

ProfileID	• 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile

ProfileType	• Identifies the technology type of the profile – 0x00 - 3GPP – 0x01 - 3GPP2

Note

- If profileID is NULL, 3GPP profile will be fetched and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be fetched

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

8.391.2 Field Documentation

8.391.2.1 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileId

8.391.2.2 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileType

8.392 pack_wds_SLQSGetRuntimeSettings_t Struct Reference

Data Fields

- uint32_t * pReqSettings

8.392.1 Detailed Description

Parameters

pReqSettings	Requested Settings (Optional Parameter) • Set bits to 1, corresponding to requested information. All other bits must be set to 0. • If the values are not available, the corresponding TLVs are not returned in the response. • Absence of this mask TLV results in the device returning all of the available information corresponding to bits 0 through 12. • In cases where the information from bit 13 or greater is required, this TLV with all the necessary bits set must be present in the request. • Values – Bit 0 - Profile identifier – Bit 1 - Profile name – Bit 2 - PDP type – Bit 3 - APN name – Bit 4 - DNS address – Bit 5 - UMTS/GPRS granted QoS – Bit 6 - Username – Bit 7 - Authentication Protocol – Bit 8 - IP address – Bit 9 - Gateway info (address and subnet mask) – Bit 10 - PCSCF address using PCO flag – Bit 11 - PCSCF server address list – Bit 12 - PCSCF domain name list – Bit 13 - MTU – Bit 14 - domain name list – Bit 15 - IP family – Bit 16 - IM_CM flag – Bit 17 - Technology name – Bit 18 - Operator reserved PCO (Not Supported on MC/EM73xx)

8.392.2 Field Documentation

8.392.2.1 uint32_t* pack_wds_SLQSGetRuntimeSettings_t::pReqSettings

8.393 pack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo curProfile

8.393.1 Detailed Description

Parameters

ProfileID	• 1 to 16 for 3GPP profile (EM/MC73xx or earlier) • 1 to 24 for 3GPP profile (EM/MC74xx onwards) • 101 to 106 for 3GPP2 profile

ProfileType	• Identifies the technology type of the profile – 0x00 - 3GPP – 0x01 - 3GPP2 – NULL is not allowed
curProfile	• union of 3GPP and 3GPP2 profile

Note

- If profileID is NULL, 3GPP profile will be created and index will be assigned based on availability in device.
- If profileID is not NULL depending on pProfileType 3GPP/3GPP2 relevant profile will be created

8.393.2 Field Documentation

8.393.2.1 wds_profileInfo pack_wds_SLQSModifyProfile_t::curProfile

8.393.2.2 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileId

8.393.2.3 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileType

8.394 pack_wds_SLQSSet3GPPConfigItem_t Struct Reference

Data Fields

- uint16_t * pLTEAttachProfile
- uint16_t * pProfileList
- uint8_t * pDefaultPDNEnabled
- uint8_t * p3gppRelease
- uint16_t * pLTEAttachProfileList
- uint16_t LTEAttachProfileListLen

8.394.1 Detailed Description

Parameters

pLTEAttach-Profile	• Optional parameter • LTE Attach Profile – points to a single WORD Value indicating the attached LTE Profile – Optional parameter with possible values 1-16 (EM/MC73xx or earlier) • This setting is deprecated on MC/EM74xx

ProfileList	Profile List • an array of 4 profile configurations • Each element points to a single WORD value indicating profile • Optional parameter with possible values – 1 - 16 (MC/EM73xx and before) – 1 - 24 (MC/EM74xx and onwards) • function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present
pDefaultPDN-Enabled	• Optional parameter – 0 - disabled – 1 - enabled
p3gppRelease	3GPP release • Optional parameter – 0 - Release_99 – 1 - Release_5 – 2 - Release_6 – 3 - Release_7 – 4 - Release_8 • In 9x30 and onwards – 5 - Release 9 – 6 - Release 10 – 7 - Release 11
pLTEAttach-ProfileList	• pointer to WORD array indicating LTE Attach Profile List – Optional parameter – possible values: 1-24 – This setting is only supported for MC/EM74xx onwards – Please provide attach profiles in order of decreasing priority in this list.
LTEAttach-ProfileListLen	• Number of element in pLTEAttachProfileList – valid range: 1-24 – This setting is only supported for MC/EM74xx onwards

8.394.2 Field Documentation

8.394.2.1 uint16_t pack_wds_SLQSSet3GPPConfigItem_t::LTEAttachProfileListLen

8.394.2.2 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::p3gppRelease

8.394.2.3 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::pDefaultPDNEnabled

8.394.2.4 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfile

8.394.2.5 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfileList

8.394.2.6 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pProfileList

8.395 pack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint8_t IPFamilyPreference

8.395.1 Detailed Description

This structure contains set IP family preference information

Parameters

IPFamily-Preference	IP Family preference • PACK_WDS_IPV4 IP Version 4 • PACK_WDS_IPV6 IP Version 6

8.395.2 Field Documentation

8.395.2.1 uint8_t pack_wds_SLQSSetIPFamilyPreference_t::IPFamilyPreference

8.396 pack_wds_SLQSSetWdsEventCallback_t Struct Reference

Data Fields

- uint8_t dataBearer
- uint8_t dormancyStatus
- uint8_t mobileIP
- uint8_t transferStats
- uint8_t currentDataBearer
- uint8_t dataSystemStatus
- uint8_t dataBearerTechExt
- uint8_t interval

8.396.1 Detailed Description

Parameters

dataBearer	data bearer
dormancyStatus	dormancy status
mobileIP	mobile IP
currentData-Bearer	current data bearer
dataSystem-Status	data system status
dataBearerTech-Ext	data Bearer Technology Extended
interval	interval

8.396.2 Field Documentation

8.396.2.1 uint8_t pack_wds_SLQSSetWdsEventCallback_t::currentDataBearer

8.396.2.2 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearer

8.396.2.3 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearerTechExt

8.396.2.4 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataSystemStatus

8.396.2.5 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dormancyStatus

8.396.2.6 uint8_t pack_wds_SLQSSetWdsEventCallback_t::interval

8.396.2.7 uint8_t pack_wds_SLQSSetWdsEventCallback_t::mobileIP

8.396.2.8 uint8_t pack_wds_SLQSSetWdsEventCallback_t::transferStats

8.397 pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4ProfileId * pProfileId

8.397.1 Detailed Description

Parameters

pProfileId	• pointer to Profile Id structure

8.397.2 Field Documentation

8.397.2.1 wdsDhcpv4ProfileId* pack_wds_SLQSSGetDHCPv4ClientConfig_t::pProfileId

8.398 pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wds_DHCPv4ProfileId * pProfileId
- wds_DHCPv4HWConfig * pHwConfig
- wds_DHCPv4OptionList * pRequestOptionList

8.398.1 Detailed Description

WDS SWI DHCPv4 Config Structure

Parameters

pProfileId	• pointer to Profile Id structure
pHWConfig	• pointer to HW Config structure

pRequestOption-List	• pointer to Option List structure to be sent in DHCP request

8.398.2 Field Documentation

8.398.2.1 wds_DHCPv4HWConfig∗ pack_wds_SLQSSSetDHCPv4ClientConfig_t::pHwConfig

8.398.2.2 wds_DHCPv4ProfileId∗ pack_wds_SLQSSSetDHCPv4ClientConfig_t::pProfileId

8.398.2.3 wds_DHCPv4OptionList∗ pack_wds_SLQSSSetDHCPv4ClientConfig_t::pRequestOptionList

8.399 pack_wds_SLQSSSetLoopback_t Struct Reference

Data Fields

- uint8_t loopbackMode
- uint8_t loopbackMultiplier

8.399.1 Detailed Description

Parameters

loopbackMode	• Loopback Mode. – 0 - Disable – 1 - Enable
loopback-Multiplier	• Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.399.2 Field Documentation

8.399.2.1 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMode

8.399.2.2 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMultiplier

8.400 pack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint8_t ∗ pTech
- uint32_t ∗ pprofileid3gpp
- uint32_t ∗ pprofileid3gpp2
- uint32_t ∗ pAuth
- char ∗ pUser
- char ∗ pPass

8.400.1 Detailed Description

Parameters

pTech	• Indicates the technology preference – 1 - UMTS – 2 - CDMA – 3 - eMBMS – 4 - Modem Link Label. Modem Link is an interface for transferring data between entities on AP and modem. • optional
pprofileid3gpp	• pointer to 3GPP profile id • optional
pprofileid3gpp2	• pointer to 3GPP2 profile id • optional
pAuth	• Authentication type, it can be PAP or CHAP • optional
pUser	• username for authentication process • optional
pPass	• password for authentication process • optional

8.400.2 Field Documentation

8.400.2.1 uint32_t* pack_wds_SLQSStartDataSession_t::pAuth

8.400.2.2 char* pack_wds_SLQSStartDataSession_t::pPass

8.400.2.3 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp

8.400.2.4 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp2

8.400.2.5 uint8_t* pack_wds_SLQSStartDataSession_t::pTech

8.400.2.6 char* pack_wds_SLQSStartDataSession_t::pUser

8.401 pack_wds_SLQSStopDataSession_t Struct Reference

Data Fields

- uint32_t * psid

8.401.1 Detailed Description

Parameters

sid	session id

8.401.2 Field Documentation

8.401.2.1 uint32_t* pack_wds_SLQSStopDataSession_t::psid

8.402 pack_wds_SLQSWdsSetEventReport_t Struct Reference

Data Fields

- uint8_t * pCurrChannelRateInd
- wds_TrStatInd * pTransferStatInd
- uint8_t * pDataBearerTechInd
- uint8_t * pDormancyStatusInd
- uint8_t * pMIPStatusInd
- uint8_t * pCurrDataBearerTechInd
- uint8_t * pDataCallStatusChangeInd
- uint8_t * pCurrPrefDataSysInd
- uint8_t * pEVDOPageMonPerChangeInd
- uint8_t * pDataSystemStatusChangeInd

8.402.1 Detailed Description

This structure contains the information about the Set Event Report Request parameters.

Parameters

pCurrChannelRateInd	(optional) • Current Channel Rate Indicator. – 0 - Do not report – 1 - Report channel rate when it changes
pTransferStatInd	(optional) • See TrStatInd for more information.
pDataBearerTechInd	(optional) • Data Bearer Technology Indicator. – 0 - Do not report – 1 - Report radio interface used for data transfer when it changes
pDormancyStatusInd	(optional) • Dormancy Status indicator. – 0 - Do not report – 1 - Report traffic channel state of interface used for data connection
pMIPStatusInd	(optional) • MIP Status Indicator. – 0 - Do not report – 1 - Report MIP status

pCurrData-BearerTechInd	(optional) • Current Data Bearer Technology Indicator. – 0 - Do not report – 1 - Report current data bearer technology when it changes
pDataCallStatus-ChangeInd	(optional) • Data Call Status Change Indicator. – 0 - Do not report – 1 - Report data call status change when it changes
pCurrPrefData-SysInd	(optional) • Current Preferred Data System Indicator. – 0 - Do not report – 1 - Report preferred data system when it changes
pEVDOPage-MonPerChange-Ind	(optional) • EV-DO Page Monitor Period Change Indicator. – 0 - Do not report – 1 - Report EV-DO page monitor period change event
pDataSystem-StatusChange-Ind	(optional) • Data System Status Change Indicator. – 0 - Do not report – 1 - Report data system status change event

Note

At least one parameter should be present.

8.402.2 Field Documentation

8.402.2.1 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrChannelRateInd

8.402.2.2 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrDataBearerTechInd

8.402.2.3 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrPrefDataSysInd

8.402.2.4 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataBearerTechInd

8.402.2.5 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataCallStatusChangeInd

8.402.2.6 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataSystemStatusChangeInd

8.402.2.7 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDormancyStatusInd

8.402.2.8 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pEVDOPageMonPerChangeInd

8.402.2.9 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pMIPStatusInd

8.402.2.10 wds_TrStatInd* pack_wds_SLQSWdsSetEventReport_t::pTransferStatInd

8.403 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference

Data Fields

- uint8_t contextId
- uint8_t contextType

8.403.1 Detailed Description

This structure contains the PDP Runtime Settings Request parameters.

Parameters

contextId	• Context Identifier
contextType	• Identifies technology type – 0 - 3GPP – 1 - 3GPP2

8.403.2 Field Documentation

8.403.2.1 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.403.2.2 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextType

8.404 PackCreateProfileOut Struct Reference

Data Fields

- uint8_t ProfileType
- uint8_t ProfileIndex
- uint16_t ExtErrorCode

8.404.1 Detailed Description

structure contains out parameter Information

Parameters

ProfileType	• Identifies the type of profile 0x00 = 3GPP 0x01 = 3GPP2
ProfileIndex	• Index identifying the profile that was created
ExtErrorCode	• The extended error code received from DS Profile subsystem

8.404.2 Field Documentation

8.404.2.1 uint16_t PackCreateProfileOut::ExtErrorCode

8.404.2.2 uint8_t PackCreateProfileOut::ProfileIndex

8.404.2.3 uint8_t PackCreateProfileOut::ProfileType

8.405 packgetDyingGaspCfg Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.405.1 Detailed Description

This struture contains the TLV required to get the Dysing GASP Config.

Parameters

pDestSMSNum[IN]	• SMS Destination Number as string of 8 bit ASCII Characters Max 20 chars. • Optional parameter.
pDestSMSContent[IN]	• SMS COntent as a string of 8 bit ASCII text characters Max 160 chars. • Optional parameter.

8.405.2 Field Documentation

8.405.2.1 uint8_t* packgetDyingGaspCfg::pDestSMSContent

8.405.2.2 uint8_t* packgetDyingGaspCfg::pDestSMSNum

8.406 packgetDyingGaspStatistics Struct Reference

Data Fields

- uint32_t * pTimeStamp
- uint8_t * pSMSAttemptedFlag

8.406.1 Detailed Description

This structure contains Get Dysing GASP Statistics

Parameters

pTimeStamp[OUT]	• Timestamp of the last time power loss was detected and Dying Gasp feature was triggered. • UTC time in seconds since Jan 06, 1980 (GPS Epoch).

pSMS-AttemptedFlag[-OUT]	• Indicates whether device attempted to send SMS in the last power loss event. • 0 - SMS not attempted • 1 - SMS attempted • This only indicates device sent the SMS but does not guarantee network delivery.

8.406.2 Field Documentation

8.406.2.1 uint8_t* packgetDyingGaspStatistics::pSMSAttemptedFlag

8.406.2.2 uint32_t* packgetDyingGaspStatistics::pTimeStamp

8.407 qmiSmsMessageList Struct Reference

Data Fields

- uint32_t messageIndex
- uint32_t messageTag

8.407.1 Detailed Description

Parameters

messageIndex	• Message index of each matched message
messageTag	• Messagetag

8.407.2 Field Documentation

8.407.2.1 uint32_t qmiSmsMessageList::messageIndex

8.407.2.2 uint32_t qmiSmsMessageList::messageTag

8.408 qmiWDSDataBearerTechnology Struct Reference

Data Fields

- uint8_t currentNetwork
- uint32_t ratMask
- uint32_t soMask

8.408.1 Detailed Description

Structure to hold the current data bearer technology values

Parameters

pCurrent-Network[OUT]	• current selected network – 0 - UNKNOWN – 1 - 3GPP2 – 2 - 3GPP
pRatMask[OUT]	• Radio Access Technology (RAT) mask to indicate the type of technology (RAT mask value of zero indicates that this field is ignored) – 0x8000 - NULL Bearer – 0x0000 - DO_NOT_CARE CDMA RAT mask – 0x01 - CDMA_1X – 0x02 - EVDO_REV0 – 0x04 - EVDO_REVA UMTS RAT mask – 0x01 - WCDMA – 0x02 - GPRS – 0x04 - HSDPA – 0x08 - HSUPA – 0x10 - EDGE – 0x20 - LTE – 0x40 - HSDPA+ – 0x80 - DC_HSDPA+
pSoMask[OUT]	• Service Option (SO) mask to indicate the SO or type of application (SO mask value of zero indicates that this field is ignored) – 0x00 - DO_NOT_CARE CDMA 1X SO mask – 0x01 - CDMA_1X_IS95 – 0x02 - CDMA_1X_IS2000 – 0x04 - CDMA_1X_IS2000_REL_A CDMA EV-DO Rev A SO mask – 0x01 - EVDO_REVA_DPA – 0x02 - EVDO_REVA_MFPA – 0x04 - EVDO_REVA_EMPA – 0x08 - EVDO_REVA_EMPA_EHRPD

8.408.2 Field Documentation

8.408.2.1 uint8_t qmiWDSDataBearerTechnology::currentNetwork

8.408.2.2 uint32_t qmiWDSDataBearerTechnology::ratMask

8.408.2.3 uint32_t qmiWDSDataBearerTechnology::soMask

8.409 RFBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBandClass

- uint16_t activeChannel

8.409.1 Detailed Description

This structure contains the RFBandInfo response parameters.

Parameters

radioInterface	• Radio interface technology – See Tables for Radio Interface
activeBandClass	• Active Band Class – See Tables for Band Classes
activeChannel	• Active channel (0 if channel is not relevant to the reported technology)

8.409.2 Field Documentation

8.409.2.1 uint16_t RFBandInfoElements::activeBandClass

8.409.2.2 uint16_t RFBandInfoElements::activeChannel

8.409.2.3 uint8_t RFBandInfoElements::radioInterface

8.410 rmTrasnferStaticsReq Struct Reference

Data Fields

- uint8_t bResetStatistics
- uint32_t ulMask

8.410.1 Detailed Description

RM Transfer Satistics Structure

Parameters

bResetStatistics	• Reset Statistics • Values: • 0 - Not Reset • Other - Reset
ulMask	• Enable/Disable RM Transfer Satatistics Indiscation Mask • Bit 0: Tx Packet Ok • Bit 1: Rx Packet Ok • Bit 2: Tx Bytes Ok • Bit 3: Rx Bytes Ok • Bit 4: Tx Packets Dropped • Bit 5: Rx Packets Dropped • Value: -0 - Disable -1 - Enable

8.410.2 Field Documentation

8.410.2.1 uint8_t rmTrasnferStaticsReq::bResetStatistics

8.410.2.2 uint32_t rmTrasnferStaticsReq::ulMask

8.411 sensorData_t Struct Reference

Data Fields

- uint32_t timeOfFirstSample
- uint8_t flags
- uint8_t sensorDataLen
- uint16_t timeOffset [64]
- uint32_t xAxis [64]
- uint32_t yAxis [64]
- uint32_t zAxis [64]

8.411.1 Detailed Description

This structure specifies information regarding the 3-Axis Sensor Data. Please check has_<Param_Name> field for presence of optional parameters

Parameters

timeOfFirst-Sample	• Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds

flags	• Flags to indicate any deviation from the default measurement assumptions. • All unused bits in this field must be set to 0. • Valid bitmasks – 0x01 - Bitmask to specify that a sign reversal is required while interpreting the sensor data; only applies to the accelerometer samples – 0x02 - Bitmask to specify that the sensor time stamp is the same as the modem time stamp
sensorDataLen	• Number of sets of the following elements – timeOffset – xAxis – yAxis – zAxis
timeOffset	• Sample time offset • Units - Milliseconds
xAxis	• Sensor x-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
yAxis	• Sensor Y-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second
xAxis	• Sensor Z-axis sample. • Units Accelerometer - Meters/seconds square • Units Gyroscope - Radians/second

8.411.2 Field Documentation

8.411.2.1 uint8_t sensorData_t::flags

8.411.2.2 uint8_t sensorData_t::sensorDataLen

8.411.2.3 uint32_t sensorData_t::timeOfFirstSample

8.411.2.4 uint16_t sensorData_t::timeOffset[64]

8.411.2.5 uint32_t sensorData_t::xAxis[64]

8.411.2.6 uint32_t sensorData_t::yAxis[64]

8.411.2.7 uint32_t sensorData_t::zAxis[64]

8.412 slot_t Struct Reference

Data Fields

- uint32_t uPhyCardStatus
- uint32_t uPhySlotStatus
- uint8_t bLogicalSlot
- uint8_t blCCIDLength
- uint8_t blCCID [255]

8.412.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

uPhyCardStatus	• State of the card in the Pyhsical Slot Status. – 0x00 - Unknown. – 0x01 - Absent. – 0x02 - Present.
uPhySlotStatus	• State of the Physical Slot status. – 0x00 Inactive. – 0x01 Activate.
bLogicalSlot	• Logical Slot associated with this physical slot. THis is valid if the physical slot is active. – 1 - Slot 1. – 2 - Slot 2. – 3 - Slot 3. – 4 - Slot 4. – 5 - Slot 5.
bLogicalSlot	• Number of sets the sets of ICCCID
blCCID[MAX_ICCID_LENGTH]	• Contains the ICCID of the card in the physical slot.

8.412.2 Field Documentation

8.412.2.1 uint8_t slot_t::blCCID[255]

8.412.2.2 uint8_t slot_t::blCCIDLength

8.412.2.3 uint8_t slot_t::bLogicalSlot

8.412.2.4 uint32_t slot_t::uPhyCardStatus

8.412.2.5 uint32_t slot_t::uPhySlotStatus

8.413 slotInf Struct Reference

Data Fields

- uint8_t cardState
- uint8_t upinState
- uint8_t upinRetries
- uint8_t upukRetries
- uint8_t errorState
- uint8_t numApp
- appStats AppStatus [10]

8.413.1 Detailed Description

This structure contains information about the SLOTS present.

Parameters

cardState	• Indicates the state of the card for each slot. – 0 - Absent – 1 - Present – 2 - Error
upinState	• Indicates the state of UPIN. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked – 0xFF - Not Available
upinRetries	• Indicates the number of retries remaining to verify the UPIN. • If 0xFF, information not available.
upukRetries	• Indicates the number of retries remaining to unblock the UPIN. • If 0xFF, information not available.

errorState	• Indicates the reason for the card error, and is valid only when the card state is Error – 0 - Unknown – 1 - Power down – 2 - Poll error – 3 - No ATR received – 4 - Volt mismatch – 5 - Parity error – 6 - Unknown; possibly removed – 7 - Card returned technical problems – 0xFF - Not Available • Other values are possible and reserved for future use. • When an unknown value is received, it is to be handled as "Unknown".
numApp	• Indicates the number of applications available on the card. • The following block is repeated for each application. i.e. AppStatus. • If zero(0) then no AppStatus information exists.
AppStatus[MAX_NO_OF_APPLICATIONS]	• See appStats for more information.

8.413.2 Field Documentation

8.413.2.1 appStats slotInf::AppStatus[10]

8.413.2.2 uint8_t slotInf::cardState

8.413.2.3 uint8_t slotInf::errorState

8.413.2.4 uint8_t slotInf::numApp

8.413.2.5 uint8_t slotInf::upinRetries

8.413.2.6 uint8_t slotInf::upinState

8.413.2.7 uint8_t slotInf::upukRetries

8.414 slots_t Struct Reference

Data Fields

- slot_t uimSlotStatus [255]

8.414.1 Field Documentation

8.414.1.1 slot_t slots_t::uimSlotStatus[255]

8.415 sms_BroadcastConfig Struct Reference

Data Fields

- uint16_t fromServiceId
- uint16_t toServiceId
- uint8_t selected

8.415.1 Detailed Description

This structure contains BroadcastConfig parameters

Parameters

fromServiceId	• Starting point of range of CBM message identifiers
toServiceId	• Ending point of range of CBM message identifiers
selected	• Range of CBM message identifiers indicated by from_service_id and to_service_id – 0x00 – Not selected – 0x01 – Selected

8.415.2 Field Documentation

8.415.2.1 uint16_t sms_BroadcastConfig::fromServiceId

8.415.2.2 uint8_t sms_BroadcastConfig::selected

8.415.2.3 uint16_t sms_BroadcastConfig::toServiceId

8.416 sms_CDMABroadcastConfig Struct Reference

Data Fields

- uint16_t serviceCategory
- uint16_t language
- uint8_t selected

8.416.1 Detailed Description

This structure contains CDMABroadcastConfig parameters

Parameters

serviceCategory	• Service category
language	• Language

selected	• Specified service_category and language – 0x00 – Not selected – 0x01 – Selected

8.416.2 Field Documentation

8.416.2.1 uint16_t sms_CDMABroadcastConfig::language

8.416.2.2 uint8_t sms_CDMABroadcastConfig::selected

8.416.2.3 uint16_t sms_CDMABroadcastConfig::serviceCategory

8.417 sms_getIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.417.1 Detailed Description

This structure contains Get Indication Register Response parameters

Parameters

pRegTrans-LayerInfoEvt	- • Optional parameter indicating registration status of transport layer information events • Values: – 0x00 - Disabled – 0x01 - Enabled • returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.
pRegTransNW-RegInfoEvt	- • Optional parameter indicating registration status of transport network registration information events • Values: – 0x00 - Disabled – 0x01 - Enabled • returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.

pRegCallStat-InfoEvt	- Optional parameter indicating registration status of call status information events
	- Values:
	  – 0x00 - Disabled
	  – 0x01 - Enabled
	- returns a default value 0xFF if this parameter is allocated memory in the structure and no response is received from the device.

8.417.2 Field Documentation

8.417.2.1 uint8_t* sms_getIndicationReg::pRegCallStatInfoEvt

8.417.2.2 uint8_t* sms_getIndicationReg::pRegTransLayerInfoEvt

8.417.2.3 uint8_t* sms_getIndicationReg::pRegTransNWRegInfoEvt

8.418 sms_getMsgWaitingInfo Struct Reference

Data Fields

- uint8_t numInstances
- sms_messageWaitingInfoContent msgWaitInfo [0xFF]

8.418.1 Detailed Description

This structure contains Get Message Waiting Info Response parameters

Parameters

numInstances	- Number of sets of the elements in structure sms_messageWaitingInfoContent
pMsgWaitInfo	- Pointer to structure of sms_messageWaitingInfoContent. – See sms_messageWaitingInfoContent for more information.

8.418.2 Field Documentation

8.418.2.1 sms_messageWaitingInfoContent sms_getMsgWaitingInfo::msgWaitInfo[0xFF]

8.418.2.2 uint8_t sms_getMsgWaitingInfo::numInstances

8.419 sms_getTransLayerInfo Struct Reference

Data Fields

- uint8_t * pRegInd
- sms_transLayerInfo * pTransLayerInfo

8.419.1 Detailed Description

This structure contains Get Transport Layer Info Response parameters

Parameters

pRegInd	- • Optional parameter indicating if transport layer is registered • Values: – 0x00 - Transport layer is not registered – 0x01 - Transport layer is registered • returns a default value 0xFF if no response is received from the device.
pTransLayerInfo	• Pointer to structure of sms_transLayerInfo. – Optional parameter – See sms_transLayerInfo for more information • returns a default value 0xFF for parameter values if no response is received from the device.

8.419.2 Field Documentation

8.419.2.1 uint8_t* sms_getTransLayerInfo::pRegInd

8.419.2.2 sms_transLayerInfo* sms_getTransLayerInfo::pTransLayerInfo

8.420 sms_getTransNWRegInfo Struct Reference

Data Fields

- uint8_t * pRegStatus

8.420.1 Detailed Description

This structure contains transport network registration info parameter

Parameters

pRegStatus	- • Optional parameter indicating transport layer network registration status • Values: – 0x00 - No service – 0x01 - In progress – 0x02 - Failed – 0x03 - Limited Service – 0x04 - Full Service • returns a default value 0xFF if no response is received from the device.

8.420.2 Field Documentation

8.420.2.1 uint8_t* sms_getTransNWRegInfo::pRegStatus

8.421 sms_maxStorageSizeReq Struct Reference

Data Fields

- uint8_t storageType
- uint8_t * pMessageMode

8.421.1 Detailed Description

This structure contains get store max size request parameters

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
pMessage-Mode(optional	parameter) • 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

Note

The Message Mode TLV must be included if the device is capable of supporting more than one protocol

8.421.2 Field Documentation

8.421.2.1 uint8_t* sms_maxStorageSizeReq::pMessageMode

8.421.2.2 uint8_t sms_maxStorageSizeReq::storageType

8.422 sms_maxStorageSizeResp Struct Reference

Data Fields

- uint32_t maxStorageSize
- uint32_t freeSlots

8.422.1 Detailed Description

This structure contains get store max size response parameters

Parameters

maxStorageSize	- • Memory Store Size

freeSlots	- Optional parameter indicating how much Memory is available - returns a default value 0xFFFFFFFF for parameter values if no response is received from the device.

8.422.2 Field Documentation

8.422.2.1 uint32_t sms_maxStorageSizeResp::freeSlots

8.422.2.2 uint32_t sms_maxStorageSizeResp::maxStorageSize

8.423 sms_messageWaitingInfoContent Struct Reference

Data Fields

- uint8_t msgType
- uint8_t activeInd
- uint8_t msgCount

8.423.1 Detailed Description

This structure contains message waiting information per instance

Parameters

msgType	- Message type
	  – 0x00 - MWI_MESSAGE_TYPE_VOICMAIL - Voicemail
	  – 0x01 - MWI_MESSAGE_TYPE_FAX - Fax
	  – 0X02 - MWI_MESSAGE_TYPE_EMAIL - Email
	  – 0x03 - MWI_MESSAGE_TYPE_OTHER - Other
	  – 0x04 - MWI_MESSAGE_TYPE_VIDEOMAIL - Videomail

activeInd	- Indicates whether the indication is active
	  – 0x00 - Inactive
	  – 0x01 - Active

msgCount	- Number of messages

8.423.2 Field Documentation

8.423.2.1 uint8_t sms_messageWaitingInfoContent::activeInd

8.423.2.2 uint8_t sms_messageWaitingInfoContent::msgCount

8.423.2.3 uint8_t sms_messageWaitingInfoContent::msgType

8.424 sms_msgProtocolResp Struct Reference

Data Fields

- uint8_t msgProtocol

8.424.1 Detailed Description

This structure contains get message protocol response parameters

Parameters

msgProtocol	- • Message Protocol • Values: – 0x00 - MESSAGE_PROTOCOL_CDMA – 0x01 - MESSAGE_PROTOCOL_WCDMA

8.424.2 Field Documentation

8.424.2.1 uint8_t sms_msgProtocolResp::msgProtocol

8.425 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_CDMABroadcastConfig CDMABroadcastConfig [0x05]

8.425.1 Detailed Description

This structure contains the 3GPP2 Broadcast Configuration Information parameters

Parameters

activated_ind	• Broadcast SMS – 0x00 - Deactivated – 0x01 - Activated
num_instances	• Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. – serviceCategory – language – selected
broadcastConfig	• A CDMABroadcastConfig structure array. • Further defined by the structure sms_CDMABroadcastConfig

8.425.2 Field Documentation

8.425.2.1 uint8_t sms_qaQmi3GPP2BroadcastCfgInfo::activated_ind

8.425.2.2 sms_CDMABroadcastConfig sms_qaQmi3GPP2BroadcastCfgInfo::CDMABroadcastConfig[0x05]

8.425.2.3 uint16_t sms_qaQmi3GPP2BroadcastCfgInfo::num_instances

8.426 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_BroadcastConfig broadcastConfig [0x05]

8.426.1 Detailed Description

This structure contains the 3GPP Broadcast Configuration Information parameters

Parameters

activated_ind	• Broadcast SMS – 0x00 - Deactivated – 0x01 - Activated
num_instances	• Number of sets (N) of parameters Following each set describes one entry in the broadcast configuration table. – fromServiceId – toServiceId – selected
broadcastConfig	• A BroadcastConfig structure array. • Further defined by the structure sms_BroadcastConfig

8.426.2 Field Documentation

8.426.2.1 uint8_t sms_qaQmi3GPPBroadcastCfgInfo::activated_ind

8.426.2.2 sms_BroadcastConfig sms_qaQmi3GPPBroadcastCfgInfo::broadcastConfig[0x05]

8.426.2.3 uint16_t sms_qaQmi3GPPBroadcastCfgInfo::num_instances

8.427 sms_routeEntry Struct Reference

Data Fields

- uint8_t messageType
- uint8_t messageClass

- uint8_t routeStorage
- uint8_t receiptAction

8.427.1 Detailed Description

This structure contains SMS route entry details

Parameters

messageType	- Message type matching this route - Values: – 0x00 - MESSAGE_TYPE_POINT_TO_POINT

messageClass	- Message Class
	- Values:
	  – 0x00 - MESSAGE_CLASS_0
	  – 0x01 - MESSAGE_CLASS_1
	  – 0x02 - MESSAGE_CLASS_2
	  – 0x03 - MESSAGE_CLASS_3
	  – 0x04 - MESSAGE_CLASS_NONE
	  – 0x05 - MESSAGE_CLASS_CDMA

routeStorage	- If the receiptAction is store where to store the message
	- Values:
	  – 0x00 - STORAGE_TYPE_UIM
	  – 0x01 - STORAGE_TYPE_NV
	  – 0xFF - STORAGE_TYPE_NONE

receiptAction	- Action to be taken on receipt of a message matching the specified type and class for this route
	- Values:
	  – 0x00 - DISCARD (discarded without notification)
	  – 0x01 - STORE AND NOTIFY (stored and notified to the registered clients)
	  – 0x02 - TRANSFER ONLY (transferred to the client, client expected to send the ACK)
	  – 0x03 - TRANSFER AND ACK (transferred to the client, device expected to send the ACK)

8.427.2 Field Documentation

8.427.2.1 uint8_t sms_routeEntry::messageClass

8.427.2.2 uint8_t sms_routeEntry::messageType

8.427.2.3 uint8_t sms_routeEntry::receiptAction

8.427.2.4 uint8_t sms_routeEntry::routeStorage

8.428 sms_sendAsyncsmsParams Struct Reference

Data Fields

- uint32_t messageFormat
- uint32_t messageSize
- uint8_t * pMessage
- uint8_t * pForceOnDC
- uint8_t * pServiceOption
- uint8_t * pFollowOnDC
- uint8_t * pLinktimer
- uint8_t * pSmsOnIms
- uint8_t * pRetryMessage
- uint32_t * pRetryMessageId
- uint32_t * pUserData

8.428.1 Detailed Description

This structure contains SMS parameters

Parameters

messageFormat	• Message format • Values: – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
messageSize	• The length of the message contents in bytes
pMessage	• The message contents
pForceOnDC	• Force the message to be sent on the CDMA dedicated channel. • Values: – 0x00 - Do not care about the channel on which the message is sent – 0x01 - Request to send the message over the dedicated channel
pServiceOption	• Service option: • Values: – 0x00 - SO_AUTO - AUTO (choose the best service option) – 0x06 - SO_6 - Service option 6 – 0x0E - SO_14 - Service option 14
pFollowOnDC	• Flag to request not to disconnect the CDMA dedicated channel after the send operation is complete. • This TLV can be included if more messages are expected to follow. • Values: – 0x01 - FOLLOW_ON_DC_ON - On (don't disconnect after send operation) Any value other than 0x01 is treated as an absence of this TLV.

pLinktimer	• Keeps the GW SMS link open for the specified number of seconds; can be enabled if more messages are expected to follow
pSmsOnIms	• Indicates whether the message is to be sent on IMS. • Values: – 0x00 - Message is not to be sent on IMS – 0x01 - Message is to be sent on IMS – 0x02 to 0xFF - Reserved
pRetryMessage	• Indicates this message is a retry message. • Values: – 0x01 - WMS_MESSAGE_IS_A_RETRY - Message is a retry message Note: Any value other than 0x01 in this field is treated as an absence of this TLV.
pRetryMessage-Id	• Message ID to be used in the retry message. • The message ID specified here is used instead of the messsage ID encoded in the raw message.
pUserData	• Enables the control point to associate the request with the corresponding indication. • The control point might send numerous requests. • This TLV will help the control point to identify the request for which the received indication belongs.

8.428.2 Field Documentation

8.428.2.1 uint32_t sms_sendAsyncsmsParams::messageFormat

8.428.2.2 uint32_t sms_sendAsyncsmsParams::messageSize

8.428.2.3 uint8_t* sms_sendAsyncsmsParams::pFollowOnDC

8.428.2.4 uint8_t* sms_sendAsyncsmsParams::pForceOnDC

8.428.2.5 uint8_t* sms_sendAsyncsmsParams::pLinktimer

8.428.2.6 uint8_t* sms_sendAsyncsmsParams::pMessage

8.428.2.7 uint8_t* sms_sendAsyncsmsParams::pRetryMessage

8.428.2.8 uint32_t* sms_sendAsyncsmsParams::pRetryMessageId

8.428.2.9 uint8_t* sms_sendAsyncsmsParams::pServiceOption

8.428.2.10 uint8_t* sms_sendAsyncsmsParams::pSmsOnIms

8.428.2.11 uint32_t* sms_sendAsyncsmsParams::pUserData

8.429 sms_setIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.429.1 Detailed Description

This structure contains Indication Register request parameters

Parameters

pRegTransLayerInfoEvt	- Optional parameter indicating registration status of transport layer information events - Values: – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
pRegTransNWRegInfoEvt	- Optional parameter indicating registration status of transport network registration information events - Values: – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device
pRegCallStatInfoEvt	- Optional parameter indicating registration status of call status information events - Values: – 0x00 - Disabled – 0x01 - Enabled – NULL - No change - specifying NULL indicates that the device will continue to use the existing setting (disable/enable) which has been previously set for the device

8.429.2 Field Documentation

8.429.2.1 uint8_t* sms_setIndicationReg::pRegCallStatInfoEvt

8.429.2.2 uint8_t* sms_setIndicationReg::pRegTransLayerInfoEvt

8.429.2.3 uint8_t* sms_setIndicationReg::pRegTransNWRegInfoEvt

8.430 sms_setRoutesReq Struct Reference

Data Fields

- uint16_t numOfRoutes

- sms_routeEntry routeList [0x0A]
- uint8_t * pTransferStatusReport

8.430.1 Detailed Description

This structure contains SMS route request parameters

Parameters

numOfRoutes	- • Number of sets of the following element
routeList	- • Array containing the set of sms_routeEntry
pTransferStatus-Report	- • 0x01 - Status report are transferred to the client (optional)

8.430.2 Field Documentation

8.430.2.1 uint16_t sms_setRoutesReq::numOfRoutes

8.430.2.2 uint8_t* sms_setRoutesReq::pTransferStatusReport

8.430.2.3 sms_routeEntry sms_setRoutesReq::routeList[0x0A]

8.431 sms_transLayerInfo Struct Reference

Data Fields

- uint8_t TransType
- uint8_t TransCap

8.431.1 Detailed Description

This structure contains Transport Layer Information

Parameters

TransType	• Transport Type – 0x00 - IMS
TransCap	• Transport Capability • Values: – 0x00 - CDMA – 0x01 - GW

8.431.2 Field Documentation

8.431.2.1 uint8_t sms_transLayerInfo::TransCap

8.431.2.2 uint8_t sms_transLayerInfo::TransType

8.432 sMSCAddress Struct Reference

Data Fields

- uint8_t length
- uint8_t data [256]

8.432.1 Detailed Description

Parameters

length	• Number of sets of following element
data	• SMSC address

8.432.2 Field Documentation

8.432.2.1 uint8_t sMSCAddress::data[256]

8.432.2.2 uint8_t sMSCAddress::length

8.433 sMSCAddressTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSCAddressInfo SMSCInfo

8.433.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
SMSCInfo	• SMSC Address • See sMSCAddressInfo for more information

8.433.2 Field Documentation

8.433.2.1 sMSCAddressInfo sMSCAddressTlv::SMSCInfo

8.433.2.2 uint8_t sMSCAddressTlv::TlvPresent

8.434 sMSEtwsMessage Struct Reference

Data Fields

- uint8_t notificationType
- uint16_t length
- uint8_t data [1254]

8.434.1 Detailed Description

Parameters

notificationType	• Message mode 0x00 - Primary 0x01 - Secondary GSM 0x02 - Secondary UMTS
length	• Number of sets of following elements
data	• Raw message data

8.434.2 Field Documentation

8.434.2.1 uint8_t sMSEtwsMessage::data[1254]

8.434.2.2 uint16_t sMSEtwsMessage::length

8.434.2.3 uint8_t sMSEtwsMessage::notificationType

8.435 sMSEtwsMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsMessageInfo EtwsMessageInfo

8.435.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
EtwsMessage-Info	• ETWS Message • See sMSEtwsMessageInfo for more information

8.435.2 Field Documentation

8.435.2.1 sMSEtwsMessageInfo sMSEtwsMessageTlv::EtwsMessageInfo

8.435.2.2 uint8_t sMSEtwsMessageTlv::TlvPresent

8.436 sMSEtwsPlmn Struct Reference

Data Fields

- uint16_t mobileCountryCode
- uint16_t mobileNetworkCode

8.436.1 Detailed Description

Parameters

mobileCountry-Code	• 16 bit representation of MCC value range : 0 -999
mobileNetwork-Code	• 16 bit representation of MNC value range : 0 -999

8.436.2 Field Documentation

8.436.2.1 uint16_t sMSEtwsPlmn::mobileCountryCode

8.436.2.2 uint16_t sMSEtwsPlmn::mobileNetworkCode

8.437 sMSMessageMode Struct Reference

Data Fields

- uint8_t messageMode

8.437.1 Detailed Description

Parameters

messageMode	Message Mode

8.437.2 Field Documentation

8.437.2.1 uint8_t sMSMessageMode::messageMode

8.438 sMSMTMessage Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex

8.438.1 Detailed Description

Parameters

storageType	memory storage 0x00-UIM 0x01-NV
messageIndex	MT Message index

8.438.2 Field Documentation

8.438.2.1 uint32_t sMSMTMessage::messageIndex

8.438.2.2 uint32_t sMSMTMessage::storageType

8.439 sMSOnIMS Struct Reference

Data Fields

- uint8_t smsOnIMS

8.439.1 Detailed Description

Parameters

smsOnIMS	SMS on IMS

8.439.2 Field Documentation

8.439.2.1 uint8_t sMSOnIMS::smsOnIMS

8.440 sMSOnIMSTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSOnIMSInfo IMSInfo

8.440.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
IMSInfo	• SMS on IMS • See sMSOnIMSInfo for more information

8.440.2 Field Documentation

8.440.2.1 sMSOnIMSInfo sMSOnIMSTlv::IMSInfo

8.440.2.2 uint8_t sMSOnIMSTlv::TlvPresent

8.441 sMSTransferRouteMTMessage Struct Reference

Data Fields

- uint8_t ackIndicator
- uint32_t transactionID
- uint8_t format
- uint16_t length
- uint8_t data [256]

8.441.1 Detailed Description

Parameters

ackIndicator	• Parameter to indicate if ACK must be sent by the control point 0x00 - Send ACK 0x01 - Do not send ACK
transactionID	• Transaction ID of the message
format	• Message format 0x00 - CDMA 0x02 - 0x05 - Reserved 0x06 - GW_PP 0x07 - GW_BC
length	• Length of the raw message. This length should not exceed the maximum WMS payload length of 256 bytes
data	• Raw message data

8.441.2 Field Documentation

8.441.2.1 uint8_t sMSTransferRouteMTMessage::ackIndicator

8.441.2.2 uint8_t sMSTransferRouteMTMessage::data[256]

8.441.2.3 uint8_t sMSTransferRouteMTMessage::format

8.441.2.4 uint16_t sMSTransferRouteMTMessage::length

8.441.2.5 uint32_t sMSTransferRouteMTMessage::transactionID

8.442 tdscdmaSigInfoExt Struct Reference

Data Fields

- float rssi
- float rscp
- float ecio
- float sinr

8.442.1 Detailed Description

This structure contains the TDSCDMA Signal Strength Info Extended

Parameters

rssi	• Measured RSSI in dB
rscp[Optional]	• Measured RSCP in dBm
ecio[Optional]	• Measured ECIO in dBm.
sinr[Optional]	• Measured SINR in dB. -15 dB is sent to clients if the actual SINR is less than -15 dB

8.442.2 Field Documentation

8.442.2.1 float tdscdmaSigInfoExt::ecio

8.442.2.2 float tdscdmaSigInfoExt::rscp

8.442.2.3 float tdscdmaSigInfoExt::rssi

8.442.2.4 float tdscdmaSigInfoExt::sinr

8.443 tempData_t Struct Reference

Data Fields

- uint32_t timeSource
- uint32_t timeOfFirstSample
- uint8_t temperatureDataLen
- uint16_t timeOffset [64]
- uint32_t temperature [64]

8.443.1 Detailed Description

This structure specifies information regarding the Temperature Data. Please check has_<Param_Name> field for presence of optional parameters

Parameters

timeSource	• Time source of the sensor data • Valid values – 0 - Sensor time source is unspecified – 1 - Time source is common between the sensors and the location engine

timeOfFirst-Sample	• Denotes a full 32-bit time stamp of the first (oldest) sample in this message. • The time stamp is in the time reference scale that is used by the sensor time source. • Units - Milliseconds
temperature-DataLen	• Number of sets of the following elements – timeOffset – temperature
timeOffset	• Sample time offset • Units - Milliseconds
temperature	• Sensor temperature. • Type - Floating point • Units - Degrees Celsius • Range -50 to +100.00

8.443.2 Field Documentation

8.443.2.1 uint32_t tempData_t::temperature[64]

8.443.2.2 uint8_t tempData_t::temperatureDataLen

8.443.2.3 uint32_t tempData_t::timeOfFirstSample

8.443.2.4 uint16_t tempData_t::timeOffset[64]

8.443.2.5 uint32_t tempData_t::timeSource

8.444 transferRouteMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSTransferRouteMTMessageInfo TransferRouteMTMessageInfo

8.444.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
TransferRouteM-TMessageInfo	• Transfer Route MT Message • See sMSTransferRouteMTMessageInfo for more information

8.444.2 Field Documentation

8.444.2.1 uint8_t transferRouteMessageTlv::TlvPresent

8.444.2.2 sMSTransferRouteMTMessageInfo transferRouteMessageTlv::TransferRouteMTMessageInfo

8.445 transferStatInd Struct Reference

Data Fields

- uint8_t StatsPeriod
- uint32_t StatsMask

8.445.1 Detailed Description

This structure contains the transfer statistic indication information

Parameters

StatsPeriod	Field Period between transfer statistic reports.
StatsMask	requested statistic bit mask.

8.445.2 Field Documentation

8.445.2.1 uint32_t transferStatInd::StatsMask

8.445.2.2 uint8_t transferStatInd::StatsPeriod

8.446 uim_appStatus Struct Reference

Data Fields

- uint8_t appType
- uint8_t appState
- uint8_t persoState
- uint8_t persoFeature
- uint8_t persoRetries
- uint8_t persoUnblockRetries
- uint8_t aidLength
- uint8_t aidVal [255]
- uint8_t univPin
- uint8_t pin1State
- uint8_t pin1Retries
- uint8_t puk1Retries
- uint8_t pin2State
- uint8_t pin2Retries
- uint8_t puk2Retries

8.446.1 Detailed Description

This structure contains Application Status Information loaded on the card.

Parameters

appType	• Indicates the type of the application. – 0 - Unknown – 1 - SIM card – 2 - USIM application – 3 - RUIM card – 4 - CSIM application – 5 - ISIM application • Other values are reserved for the future and are to be handled as "Unknown".
appState	• Indicates the state of the application. – 0 - Unknown – 1 - Detected – 2 - PIN1 or UPIN is required – 3 - PUK1 or PUK for UPIN is required – 4 - Personalization state must be checked – 5 - PIN1 is blocked – 6 - Illegal – 7 - Ready
persoState	• Indicates the state of the personalization for the application. – 0 - Unknown – 1 - Personalization operation is in progress – 2 - Ready – 3 - Personalization code is required – 4 - PUK for personalization code is required – 5 - Permanently blocked
persoFeature	• Indicates the personalization feature. • This applies only when a personalization code is required to deactivate or unblock personalization. – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization – 11 - Unknown
persoRetries	• Indicates the number of retries remaining to disable the personalization.

persoUnblock-Retries	• Indicates the number of retries remaining to unblock the personalization.
aidLength	• Number of sets of the following elements. i.e. aidVal • If zero(0) then no aidVal information exists.
aidVal[MAX_DE-SCRIPTION_LE-NGTH]	• Application identifier value.
univPin	• Indicates whether UPIN replaces PIN1. – 0 - PIN1 is used – 1 - UPIN replaces PIN1
pin1State	• Indicates the state of PIN1. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin1Retries	• Indicates the number of retries remaining to verify PIN1.
puk1Retries	• Indicates the number of retries remaining to unblock PIN1.
pin2State	• Indicates the state of PIN2. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked
pin2Retries	• Indicates the number of retries remaining to verify PIN2.
puk2Retries	• Indicates the number of retries remaining to unblock PIN2.

8.446.2 Field Documentation

8.446.2.1 uint8_t uim_appStatus::aidLength

8.446.2.2 uint8_t uim_appStatus::aidVal[255]

8.446.2.3 uint8_t uim_appStatus::appState

8.446.2.4 uint8_t uim_appStatus::appType

8.446.2.5 uint8_t uim_appStatus::persoFeature

8.446.2.6 uint8_t uim_appStatus::persoRetries

8.446.2.7 uint8_t uim_appStatus::persoState

8.446.2.8 uint8_t uim_appStatus::persoUnblockRetries

8.446.2.9 uint8_t uim_appStatus::pin1Retries

8.446.2.10 uint8_t uim_appStatus::pin1State

8.446.2.11 uint8_t uim_appStatus::pin2Retries

8.446.2.12 uint8_t uim_appStatus::pin2State

8.446.2.13 uint8_t uim_appStatus::puk1Retries

8.446.2.14 uint8_t uim_appStatus::puk2Retries

8.446.2.15 uint8_t uim_appStatus::univPin

8.447 uim_authenticateResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [1024]

8.447.1 Detailed Description

This structure contains the information about the authenticate result.

Parameters

contentLen	• Length of the following elements i.e. content.
content[UIM_MAX_CONTENT_LENGTH]	• Authenticate data. • This value is a sequence of bytes returned from the card.

8.447.2 Field Documentation

8.447.2.1 uint8_t uim_authenticateResult::content[1024]

8.447.2.2 uint16_t uim_authenticateResult::contentLen

8.448 uim_authenticationData Struct Reference

Data Fields

- uint8_t context
- uint16_t dataLen
- uint8_t data [1024]

8.448.1 Detailed Description

This structure contains the Session Information.

Parameters

context	• Authenticate context. – 0 - Runs the GSM algorithm (valid only on a 2GSIM card) – 1 - Runs the CAVE algorithm (valid only on a RUIM card) – 2 - GSM security context (valid only on a USIM application) – 3 - 3G security context (valid only on a USIM application) – 4 - VGCS/VBS security context (valid only on aUSIM application) – 5 - GBA security context, Bootstrapping mode (valid only on a USIM or ISIM application) – 6 - GBA security context, NAF Derivation mode (valid only on a USIM or ISIM application) – 7 - MBMS security context, MSK Update mode (valid only on a USIM application) – 8 - MBMS security context, MTK Generation mode (valid only on a USIM application) – 9 - MBMS security context, MSK Deletion mode (valid only on a USIM application) – 10 - MBMS security context, MUK Deletion mode (valid only on a USIM application) – 11 - IMS AKA security context (valid only on aISIM application) – 12 - HTTP-digest security context (valid only onan ISIM application) – 13 - Compute IP authentication, CHAP (valid onlyon RUIM or CSIM) – 14 - Compute IP authentication, MN-HA authenticator (valid only on RUIM or CSIM) – 15 - Compute IP authentication, MIP-RRQ hash (valid only on RUIM or CSIM) – 16 - Compute IP authentication, MN-AAA authenticator (valid only on RUIM or CSIM) – 17 - Compute IP authentication, HRPD access authenticator (valid only on RUIM or CSIM) • Other values are possible and reserved for future use.
dataLen	• Length of the following elements i.e. data.
data[UIM_MAX-_DESCRIPTIO-N_LENGTH]	• Authenticate Data.

8.448.2 Field Documentation

8.448.2.1 uint8_t uim_authenticationData::context

8.448.2.2 uint8_t uim_authenticationData::data[1024]

8.448.2.3 uint16_t uim_authenticationData::dataLen

8.449 uim_cardResult Struct Reference

Data Fields

- uint8_t sw1
- uint8_t sw2

8.449.1 Detailed Description

This structure contains the information about the card result.

Parameters

sw1	• SW1 received from the card.
sw2	• SW2 received from the card.

8.449.2 Field Documentation

8.449.2.1 uint8_t uim_cardResult::sw1

8.449.2.2 uint8_t uim_cardResult::sw2

8.450 uim_cardStatus Struct Reference

Data Fields

- uint16_t indexGwPri
- uint16_t index1xPri
- uint16_t indexGwSec
- uint16_t index1xSec
- uint8_t numSlot
- uim_slotInfo SlotInfo [5]

8.450.1 Detailed Description

This structure contains Card Status Information.

Parameters

indexGwPri	• Index of the primary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.

index1xPri	• Index of the primary 1X provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
indexGwSec	• Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
index1xSec	• Index of the secondary GW provisioning application. • The most significant byte indicates the slot (starting from 0), while the least significant byte indicates the application for that slot (starting from 0). • The value 0xFFFF identifies when the session does not exist.
numSlot	• Indicates the number of slots available on the device. • The following block is repeated for each slot. i.e. cardState • If zero(0) then no cardState information exists.
SlotInfo	• See uim_slotInfo for more information.

8.450.2 Field Documentation

8.450.2.1 uint16_t uim_cardStatus::index1xPri

8.450.2.2 uint16_t uim_cardStatus::index1xSec

8.450.2.3 uint16_t uim_cardStatus::indexGwPri

8.450.2.4 uint16_t uim_cardStatus::indexGwSec

8.450.2.5 uint8_t uim_cardStatus::numSlot

8.450.2.6 uim_slotInfo uim_cardStatus::SlotInfo[5]

8.451 uim_changeUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t oldPINLen
- uint8_t oldPINVal [255]
- uint8_t pinLen
- uint8_t pinVal [255]

8.451.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

pinID	• Indicates the PIN ID to be changed. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN – 4 - Hidden key
oldPINLen	• Length of the following elements i.e. old pin value.
oldPINVal[MAX_DESCRIPTION_LENGTH]	• Old PIN value. • This value is a sequence of ASCII characters.
pinLen	• Length of the following elements i.e. new pin value.
pinVal[MAX_DESCRIPTION_LENGTH]	• New PIN value. • This value is a sequence of ASCII characters.

8.451.2 Field Documentation

8.451.2.1 uint8_t uim_changeUIMPIN::oldPINLen

8.451.2.2 uint8_t uim_changeUIMPIN::oldPINVal[255]

8.451.2.3 uint8_t uim_changeUIMPIN::pinID

8.451.2.4 uint8_t uim_changeUIMPIN::pinLen

8.451.2.5 uint8_t uim_changeUIMPIN::pinVal[255]

8.452 uim_depersonalizationInformation Struct Reference

Data Fields

- uint8_t feature
- uint8_t operation
- uint8_t ckLen
- uint8_t ckVal [255]

8.452.1 Detailed Description

This structure contains the Depersonalization Information.

Parameters

feature	• Indicates the personalization feature to de-activate or unblock. – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization
operation	• Indicates the operation to perform. – 0 - Deactivate personalization. – 1 - Unblock personalization.
ckLen	• Length of the following elements i.e. control key value.
ckVal[MAX_DESCRIPTION_LENGTH]	• Control key value. • This value is a sequence of ASCII characters.

8.452.2 Field Documentation

8.452.2.1 uint8_t uim_depersonalizationInformation::ckLen

8.452.2.2 uint8_t uim_depersonalizationInformation::ckVal[255]

8.452.2.3 uint8_t uim_depersonalizationInformation::feature

8.452.2.4 uint8_t uim_depersonalizationInformation::operation

8.453 uim_encryptedPIN1 Struct Reference

Data Fields

- uint8_t pin1Len
- uint8_t pin1Val [255]

8.453.1 Detailed Description

This structure contains the encrypted PIN1 Information.

Parameters

pin1Len	• Number of sets of the following elements ie encrypted PIN1 value. • If zero(0), no information follows.
pin1Val	• Encrypted PIN1 value.

Note

This value is returned only when PIN1 is enabled successfully and the feature is supported.

8.453.2 Field Documentation

8.453.2.1 uint8_t uim_encryptedPIN1::pin1Len

8.453.2.2 uint8_t uim_encryptedPIN1::pin1Val[255]

8.454 uim_fileAttributes Struct Reference

Data Fields

- uint16_t fileSize
- uint16_t fileID
- uint8_t fileType
- uint16_t recordSize
- uint16_t recordCount
- uint8_t secRead
- uint16_t secReadMask
- uint8_t secWrite
- uint16_t secWriteMask
- uint8_t secIncrease
- uint16_t secIncreaseMask
- uint8_t secDeactivate
- uint16_t secDeactivateMask
- uint8_t secActivate
- uint16_t secActivateMask
- uint16_t rawLen
- uint8_t rawValue [255]

8.454.1 Detailed Description

This structure contains the information about the File Attributes.

Parameters

fileSize	• Indicates the size of the file.
fileID	• Indicates the ID of the file.

fileType	• Indicates the type of the file. – 0 - Transparent – 1 - Cyclic – 2 - Linear fixed – 3 - Dedicated file – 4 - Master file
recordSize	• Indicates the size of the records. • Only for cyclic and linear fixed files
recordCount	• Indicates the total no. of the records. • Only for linear fixed files
secRead	• Read security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secReadMask	• Mask with read security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
secWrite	• Write security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secWriteMask	• Mask with write security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM

secIncrease	• Increase security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secIncrease-Mask	• Mask with increase security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
secDeactivate	• Deactivate security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secDeactivate-Mask	• Mask with deactivate security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
secActivate	• Activate security attributes. – 0 - Always – 1 - Never – 2 - AND condition – 3 - OR condition – 4 - Single condition
secActivateMask	• Mask with activate security attributes. • This field is valid only when required by security attributes. – Bit 0 - PIN1 – Bit 1 - PIN2 – Bit 2 - UPIN – Bit 3 - ADM
rawLen	• Length of the following elements i.e. raw value.

rawValue[MAX_-DESCRIPTION-_LENGTH]	• Raw value of file attributes.

8.454.2 Field Documentation

8.454.2.1 uint16_t uim_fileAttributes::fileID

8.454.2.2 uint16_t uim_fileAttributes::fileSize

8.454.2.3 uint8_t uim_fileAttributes::fileType

8.454.2.4 uint16_t uim_fileAttributes::rawLen

8.454.2.5 uint8_t uim_fileAttributes::rawValue[255]

8.454.2.6 uint16_t uim_fileAttributes::recordCount

8.454.2.7 uint16_t uim_fileAttributes::recordSize

8.454.2.8 uint8_t uim_fileAttributes::secActivate

8.454.2.9 uint16_t uim_fileAttributes::secActivateMask

8.454.2.10 uint8_t uim_fileAttributes::secDeactivate

8.454.2.11 uint16_t uim_fileAttributes::secDeactivateMask

8.454.2.12 uint8_t uim_fileAttributes::secIncrease

8.454.2.13 uint16_t uim_fileAttributes::secIncreaseMask

8.454.2.14 uint8_t uim_fileAttributes::secRead

8.454.2.15 uint16_t uim_fileAttributes::secReadMask

8.454.2.16 uint8_t uim_fileAttributes::secWrite

8.454.2.17 uint16_t uim_fileAttributes::secWriteMask

8.455 uim_fileInfo Struct Reference

Data Fields

- uint16_t fileID
- uint8_t pathLen
- uint16_t path [255]

8.455.1 Detailed Description

This structure contains paramaters for file Information

Parameters

fileID	• This is Identifier to SIM files; e.g. in UIM "6F07" is Identifier of IMSI File
pathLen	• Length of file Path
path	• Path value. This value must be the complete path of the file, which is a sequence block of 2 bytes (e.g., 0x3F00 0x7FFF).

8.455.2 Field Documentation

8.455.2.1 uint16_t uim_fileInfo::fileID

8.455.2.2 uint16_t uim_fileInfo::path[255]

8.455.2.3 uint8_t uim_fileInfo::pathLen

8.456 uim_hotSwapStatus Struct Reference

Data Fields

- uint8_t hotSwapLength
- uint8_t hotSwap [255]

8.456.1 Detailed Description

This structure contains Hot Swap Status Information.

Parameters

hotSwapLength	• Number of sets of the following elements. i.e. hot_swap
hotSwap	• Indicates the status of the hot-swap switch. – 0 - Hot-swap is not supported – 1 - Hot-swap is supported, but the status of the switch is not supported – 2 - Switch indicates that the card is present – 3 - Switch indicates that the card is not present

8.456.2 Field Documentation

8.456.2.1 uint8_t uim_hotSwapStatus::hotSwap[255]

8.456.2.2 uint8_t uim_hotSwapStatus::hotSwapLength

8.457 uim_personalizationStatus Struct Reference

Data Fields

- uint8_t numFeatures
- uint8_t feature [12]
- uint8_t verifyLeft [12]
- uint8_t unblockLeft [12]

8.457.1 Detailed Description

This structure contains the information about the card result.

Parameters

numFeatures	• Number of active personalization features. The following block is repeated for each feature.
feature	• Indicates the personalization feature to deactivate or unblock. Valid values: – 0 - GW network personalization – 1 - GW network subset personalization – 2 - GW service provider personalization – 3 - GW corporate personalization – 4 - GW UIM personalization – 5 - 1X network type 1 personalization – 6 - 1X network type 2 personalization – 7 - 1X HRPD personalization – 8 - 1X service provider personalization – 9 - 1X corporate personalization – 10 - 1X RUIM personalization
verifyLeft	• Number of the remaining attempts to verify the personalization feature.
unblockLeft	• Number of the remaining attempts to unblock the personalization feature.

8.457.2 Field Documentation

8.457.2.1 uint8_t uim_personalizationStatus::feature[12]

8.457.2.2 uint8_t uim_personalizationStatus::numFeatures

8.457.2.3 uint8_t uim_personalizationStatus::unblockLeft[12]

8.457.2.4 uint8_t uim_personalizationStatus::verifyLeft[12]

8.458 uim_readResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [255]

8.458.1 Detailed Description

This structure contains the information for write operation.

Parameters

contentLen	• Number of sets of content.
content[255]	• Read content. • The content is the sequence of bytes as read from the card.

8.458.2 Field Documentation

8.458.2.1 uint8_t uim_readResult::content[255]

8.458.2.2 uint16_t uim_readResult::contentLen

8.459 uim_readTransparentInfo Struct Reference

Data Fields

- uint16_t offset
- uint16_t length

8.459.1 Detailed Description

This structure contains the information for read operation.

Parameters

offset	• Offset for the read operation.
length	• Length of the content to be read. • The value 0 is used to read the complete file.

8.459.2 Field Documentation

8.459.2.1 uint16_t uim_readTransparentInfo::length

8.459.2.2 uint16_t uim_readTransparentInfo::offset

8.460 uim_refreshevent Struct Reference

Data Fields

- uint8_t stage
- uint8_t mode

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]
- uint16_t numOfFiles
- uim_fileInfo arrfileInfo [255]

8.460.1 Detailed Description

This structure contains information of parameters associated with the Refresh Event.

Parameters

stage	• Indicates the stage of the Refresh procedure. – 0 - Waiting for OK to refresh – 1 - Refresh started – 2 - Refresh ended successfully – 3 - Refresh failed
mode	• Indicates the Refresh mode. – 0 - Reset – 1 - Init – 2 - Init and FCN – 3 - FCN – 4 - Init and Full FCN – 5 - Application reset – 6 - 3G session reset
sessionType	• Indicates the session type. – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Nonprovisioning on slot 1 – 5 - Nonprovisioning on slot 2 – 6 - Card on slot 1 – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2
aidLength	• Number of sets of the following elements – Application Identifier
aid	• Application identifier value or channel ID. This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases

numFiles	• Number of sets of the following elements: – file_id – path_len – path
arrfileInfo	• Array of file Information struct

8.460.2 Field Documentation

8.460.2.1 uint8_t uim_refreshevent::aid[255]

8.460.2.2 uint8_t uim_refreshevent::aidLength

8.460.2.3 uim_fileInfo uim_refreshevent::arrfileInfo[255]

8.460.2.4 uint8_t uim_refreshevent::mode

8.460.2.5 uint16_t uim_refreshevent::numOfFiles

8.460.2.6 uint8_t uim_refreshevent::sessionType

8.460.2.7 uint8_t uim_refreshevent::stage

8.461 uim_registerRefresh Struct Reference

Data Fields

- uint8_t registerFlag
- uint8_t voteForInit
- uint16_t numFiles
- uim_fileInfo arrfileInfo [255]

8.461.1 Detailed Description

This structure contains parameters of refresh Information

Parameters

registerFlag	• Flag that indicates whether to register or deregister for refresh indications. Valid values: – 0 - Deregister – 1 - Register
voteForInit	• Flag that indicates whether to vote for the init when there is a refresh. Valid values: – 0 - Client does not vote for initialization – 1 - Client votes for initialization

numFiles	• Number of sets of the following elements: – file_id – path_len – path
arrfileInfo	• Array of file Information structure. • See uim_fileInfo for more information

8.461.2 Field Documentation

8.461.2.1 uim_fileInfo uim_registerRefresh::arrfileInfo[255]

8.461.2.2 uint16_t uim_registerRefresh::numFiles

8.461.2.3 uint8_t uim_registerRefresh::registerFlag

8.461.2.4 uint8_t uim_registerRefresh::voteForInit

8.462 uim_remainingRetries Struct Reference

Data Fields

- uint8_t verifyLeft
- uint8_t unblockLeft

8.462.1 Detailed Description

This structure contains the information about the retries remaining.

Parameters

verifyLeft	• Number of remaining attempts to verify the PIN. • 0xFF, if unavailable.
unblockLeft	• Number of remaining attempts to unblock the PIN. • 0xFF, if unavailable.

Note

This value is returned only when the enable/disable operation has failed. This information is not sent for a hidden key PIN type.

8.462.2 Field Documentation

8.462.2.1 uint8_t uim_remainingRetries::unblockLeft

8.462.2.2 uint8_t uim_remainingRetries::verifyLeft

8.463 uim_sessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.463.1 Detailed Description

This structure contains the Session Information.

Parameters

sessionType	• Indicates the session type. – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Non-provisioning on slot 1 – 5 - Non-provisioning on slot 2 – 6 - Card on slot 1 – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2
aidLength	• Length of the following elements i.e. Application Identifier.
aid	• Application identifier value or channel ID. • This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases.

8.463.2 Field Documentation

8.463.2.1 uint8_t uim_sessionInformation::aid[255]

8.463.2.2 uint8_t uim_sessionInformation::aidLength

8.463.2.3 uint8_t uim_sessionInformation::sessionType

8.464 uim_setPINProtection Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinOperation

- uint8_t pinLength
- uint8_t pinValue [255]

8.464.1 Detailed Description

This structure contains the information about the pin protection parameters that need to be set.

Parameters

pinID	• Indicates the PIN ID to be enabled or disabled. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN – 4 - Hidden key
pinOperation	• Indicates whether the PIN is enabled or disabled. – 0 - Disable the PIN – 1 - Enable the PIN
pinLength	• Length of the following elements i.e. pin value.
pinValue[MAX_-DESCRIPTION-_LENGTH]	• PIN value. • This value is a sequence of ASCII characters.

8.464.2 Field Documentation

8.464.2.1 uint8_t uim_setPINProtection::pinID

8.464.2.2 uint8_t uim_setPINProtection::pinLength

8.464.2.3 uint8_t uim_setPINProtection::pinOperation

8.464.2.4 uint8_t uim_setPINProtection::pinValue[255]

8.465 uim_slotInfo Struct Reference

Data Fields

- uint8_t cardState
- uint8_t upinState
- uint8_t upinRetries
- uint8_t upukRetries
- uint8_t errorState
- uint8_t numApp
- uim_appStatus AppStatus [10]

8.465.1 Detailed Description

This structure contains information about the SLOTS present.

Parameters

cardState	• Indicates the state of the card for each slot. – 0 - Absent – 1 - Present – 2 - Error
upinState	• Indicates the state of UPIN. – 0 - Unknown – 1 - Enabled and not verified – 2 - Enabled and verified – 3 - Disabled – 4 - Blocked – 5 - Permanently blocked – 0xFF - Not Available
upinRetries	• Indicates the number of retries remaining to verify the UPIN. • If 0xFF, information not available.
upukRetries	• Indicates the number of retries remaining to unblock the UPIN. • If 0xFF, information not available.
errorState	• Indicates the reason for the card error, and is valid only when the card state is Error – 0 - Unknown – 1 - Power down – 2 - Poll error – 3 - No ATR received – 4 - Volt mismatch – 5 - Parity error – 6 - Unknown; possibly removed – 7 - Card returned technical problems – 0xFF - Not Available • Other values are possible and reserved for future use. • When an unknown value is received, it is to be handled as "Unknown".
numApp	• Indicates the number of applications available on the card. • The following block is repeated for each application. i.e. AppStatus. • If zero(0) then no AppStatus information exists.
AppStatus	• See uim_appStatus for more information.

8.465.2 Field Documentation

8.465.2.1 uim_appStatus uim_slotInfo::AppStatus[10]

8.465.2.2 uint8_t uim_slotInfo::cardState

8.465.2.3 uint8_t uim_slotInfo::errorState

8.465.2.4 uint8_t uim_slotInfo::numApp

8.465.2.5 uint8_t uim_slotInfo::upinRetries

8.465.2.6 uint8_t uim_slotInfo::upinState

8.465.2.7 uint8_t uim_slotInfo::upukRetries

8.466 uim_UIMSessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.466.1 Detailed Description

This structure contains the Session Information.

Parameters

sessionType	• Indicates the session type. – 0 - Primary GW provisioning – 1 - Primary 1X provisioning – 2 - Secondary GW provisioning – 3 - Secondary 1X provisioning – 4 - Non-provisioning on slot 1 – 5 - Non-provisioning on slot 2 – 6 - Card on slot 1 – 7 - Card on slot 2 – 8 - Logical channel on slot 1 – 9 - Logical channel on slot 2
aidLength	• Length of the following elements i.e. Application Identifier.
aid	• Application identifier value or channel ID. • This value is required for non-provisioning and for logical channel session types. It is ignored in all other cases.

8.466.2 Field Documentation

8.466.2.1 uint8_t uim_UIMSessionInformation::aid[255]

8.466.2.2 uint8_t uim_UIMSessionInformation::aidLength

8.466.2.3 uint8_t uim_UIMSessionInformation::sessionType

8.467 uim_unblockUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pukLen
- uint8_t pukVal [255]
- uint8_t newPINLen
- uint8_t newPINVal [255]

8.467.1 Detailed Description

This structure contains the information about the unblock pin parameters.

Parameters

pinID	• Indicates the PIN ID to be changed. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN
pukLen	• Length of the following elements i.e. puk value.
pukVal[UIM_MAX_DESCRIPTION_LENGTH]	• PIN Unlock Key value. • This value is a sequence of ASCII characters.
newPINLen	• Length of the following elements i.e. new pin value.
newPINVal[UIM_MAX_DESCRIPTION_LENGTH]	• New PIN value. • This value is a sequence of ASCII characters.

8.467.2 Field Documentation

8.467.2.1 uint8_t uim_unblockUIMPIN::newPINLen

8.467.2.2 uint8_t uim_unblockUIMPIN::newPINVal[255]

8.467.2.3 uint8_t uim_unblockUIMPIN::pinID

8.467.2.4 uint8_t uim_unblockUIMPIN::pukLen

8.467.2.5 uint8_t uim_unblockUIMPIN::pukVal[255]

8.468 uim_verifyUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinLen
- uint8_t pinVal [255]

8.468.1 Detailed Description

This structure contains the information about the pin parameters that need to be verified.

Parameters

pinID	• Indicates the PIN ID to be verified. – 1 - PIN1 (also called PIN) – 2 - PIN2 – 3 - Universal PIN – 4 - Hidden key
pinLen	• Length of the following elements i.e. pin value.
pinVal[MAX_DESCRIPTION_LENGTH]	• PIN value. • This value is a sequence of ASCII characters.

8.468.2 Field Documentation

8.468.2.1 uint8_t uim_verifyUIMPIN::pinID

8.468.2.2 uint8_t uim_verifyUIMPIN::pinLen

8.468.2.3 uint8_t uim_verifyUIMPIN::pinVal[255]

8.469 unpack_cat_SetCatEventCallback_ind_t Struct Reference

Data Fields

- uint8_t event_Index
- struct cat_commonEventTlv CCETlv [11]

8.469.1 Detailed Description

Structure used to store all CAT Notification Parameters.

Parameters

event_Index	- Event Index
CCETlv	- CAT common event Tlv cat_commonEventTlv for more info.

8.469.2 Field Documentation

8.469.2.1 struct cat_commonEventTlv unpack_cat_SetCatEventCallback_ind_t::CCETlv[11]

8.469.2.2 uint8_t unpack_cat_SetCatEventCallback_ind_t::event_Index

8.470 unpack_cat_SetCATEventCallback_t Struct Reference

Data Fields

- uint32_t errorMask
- uint16_t Tlvresult

8.470.1 Detailed Description

This structure contains sEnables/disables the CAT event callback unpack variable.

Parameters

errorMask	• error bitmask. Each bit set indicates the proactive command that caused the error – 0x00000001 - Display Text – 0x00000002 - Get In-Key – 0x00000004 - Get Input – 0x00000008 - Setup Menu – 0x00000010 - Select Item – 0x00000020 - Send SMS - Alpha Identifier – 0x00000040 - Setup Event: User Activity – 0x00000080 - Setup Event: Idle Screen Notify – 0x00000100 - Setup Event: Language Sel Notify – 0x00000200 - Setup Idle Mode Text – 0x00000400 - Language Notification – 0x00000800 - Refresh – 0x00001000 - End Proactive Session
Tlvresult	• Unpack Result

8.470.2 Field Documentation

8.470.2.1 uint32_t unpack_cat_SetCATEventCallback_t::errorMask

8.470.2.2 uint16_t unpack_cat_SetCATEventCallback_t::Tlvresult

8.471 unpack_dms_GetActivationState_t Struct Reference

Data Fields

- uint8_t state

8.471.1 Detailed Description

This structure contains unpack_dms_GetActivationState parameters.

Parameters

state[OUT]	• Service Activation Code 0 - Service not activated 1 - Service activated 2 - Activation connecting 3 - Activation connected 4 - OTASP security authenticated 5 - OTASP NAM downloaded 6 - OTASP MDN downloaded 7 - OTASP IMSI downloaded 8 - OTASP PRL downloaded 9 - OTASP SPC downloaded 10 - OTASP settings committed

8.471.2 Field Documentation

8.471.2.1 uint8_t unpack_dms_GetActivationState_t::state

8.472 unpack_dms_GetBandCapability_t Struct Reference

Data Fields

- uint32_t BandCapability
- uint16_t Tlvresult

8.472.1 Detailed Description

Structure to store the band capability of the device.

Parameters

BandCapability	Bitmask of bands supported by the device • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19
Tlvresult	• unpack Tlv Result.

8.472.2 Field Documentation

8.472.2.1 uint32_t unpack_dms_GetBandCapability_t::BandCapability

8.472.2.2 uint16_t unpack_dms_GetBandCapability_t::Tlvresult

8.473 unpack_dms_GetCrashAction_t Struct Reference

Data Fields

- uint8_t DevCrashState
- uint16_t Tlvresult

8.473.1 Detailed Description

This structure is used to store the Crash State from the device.

Parameters

DevCrashState	• Device Crash State • Values: – 0 - USB Memory Download Modem will reset after a crash and will stay in USB download mode with only ttyUSB0 enumerated. ramdump tool is to be used to recover the crash dump. Modem needs to be reset again to come back in ONLINE mode. – 1 - Reset Modem will reset and come back in ONLINE mode. Minimal crash data will be available and can be extracted with at!gcdump? AT command or SLQSSwiGetCrashInfo() SDK API – 2 - No action
Tlvresult	• unpack Tlv result

8.473.2 Field Documentation

8.473.2.1 uint8_t unpack_dms_GetCrashAction_t::DevCrashState

8.473.2.2 uint16_t unpack_dms_GetCrashAction_t::Tlvresult

8.474 unpack_dms_GetCustFeature_t Struct Reference

Data Fields

- uint32_t GpsEnable
- uint8_t DisableIMSI
- uint16_t IPFamSupport
- uint8_t RMAutoConnect
- uint8_t GPSSel
- uint8_t SMSSupport
- uint8_t IsVoiceEnabled
- uint8_t DHCPRelayEnabled
- uint8_t GPSLPM
- uint16_t Tlvresult

8.474.1 Detailed Description

This structure contains current settings of custom features

Parameters

GpsEnable	• describes if GPS is enabled or disabled • values: – 0x00 - GPS is disabled – 0x01 - GPS is enabled
DisableIMSI	• optional 1 byte parameter • describes if IMSI display is enabled or disabled • values: – 0x00 - Allow display of IMSI – 0x01 - Do not display IMSI
IPFamSupport	• optional 2 byte BitMask • bitmask representing the IP families supported • values: – 0x01 - IPv4 – 0x02 - IPv6 – 0x04 - IPv4v6
RMAutoConnect	• optional 1 byte parameter • QMI Mode RM Net Auto Connect Support • values: – 0x00 - Not Supported – 0x01 - Supported
GPSSel	• optional 1 byte parameter • GPS Antenna Select • values: – 0x00 - Dedicated GPS Port – 0x01 - GPS Rx over AUX Port – 0x02 - GPS Rx over dedicated GPS port with no bias voltage applied
SMSSupport	• optional 1 byte parameter • SMS support • values: – 0x00 - Not supported – 0x01 - supported • Used to determine whether or not to hide SMS from user

IsVoiceEnabled	• optional 1 byte parameter • Voice support • values: – 0x00 - Enable voice on both AT and QMI interface (default) – 0x01 - Reserved – 0x02 - Disable voice on both AT and QMI interface
DHCPRelay-Enabled	• optional 1 byte parameter • DHCP Relay support • values: – 0x00 - Disable DHCP relay – 0x01 - Enable DHCP relay
GPSLPM	• optional 1 byte parameter • GPSLPM support • values: – 0x00 - Enable GPS in Low Power Mode – 0x01 - Disable GPS in Low Power Mode

8.474.2 Field Documentation

8.474.2.1 uint8_t unpack_dms_GetCustFeature_t::DHCPRelayEnabled

8.474.2.2 uint8_t unpack_dms_GetCustFeature_t::DisableIMSI

8.474.2.3 uint32_t unpack_dms_GetCustFeature_t::GpsEnable

8.474.2.4 uint8_t unpack_dms_GetCustFeature_t::GPSLPM

8.474.2.5 uint8_t unpack_dms_GetCustFeature_t::GPSSel

8.474.2.6 uint16_t unpack_dms_GetCustFeature_t::IPFamSupport

8.474.2.7 uint8_t unpack_dms_GetCustFeature_t::IsVoiceEnabled

8.474.2.8 uint8_t unpack_dms_GetCustFeature_t::RMAutoConnect

8.474.2.9 uint8_t unpack_dms_GetCustFeature_t::SMSSupport

8.474.2.10 uint16_t unpack_dms_GetCustFeature_t::Tlvresult

8.475 unpack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- DMSgetCustomFeatureV2 GetCustomFeatureV2
- uint16_t Tlvresult

8.475.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

GetCustomFeatureV2	• See DMSgetCustomFeatureV2 for more information
Tlvresult	• Unpack Result

8.475.2 Field Documentation

8.475.2.1 DMSgetCustomFeatureV2 unpack_dms_GetCustFeaturesV2_t::GetCustomFeatureV2

8.475.2.2 uint16_t unpack_dms_GetCustFeaturesV2_t::Tlvresult

8.476 unpack_dms_GetDeviceCap_t Struct Reference

Data Fields

- uint32_t MaxTXChannelRate
- uint32_t MaxRXChannelRate
- uint32_t DataServiceCapability
- uint32_t SimCapability
- uint32_t RadioIfacesSize
- uint8_t RadioIfaces [64]
- uint16_t Tlvresult

8.476.1 Detailed Description

Structure to store the device capabilities

Parameters

MaxTXChannelRate	• Maximum transmission rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies
MaxRXChannelRate	• Maximum reception rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies
DataServiceCapability	• CS/PS data service capability – 0 - No data services supported – 1 - Only Circuit Switched (CS) services supported – 2 - Only Packet Switched (PS) services supported – 3 - Simultaneous CS and PS – 4 - Non-simultaneous CS and PS

SimCapability	Device SIM capability • 0 - SIM not supported • 1 - SIM supported
RadioIfacesSize	• Upon input, the maximum number of elements that the radio interface array can contain • Upon successful output, actual number of elements in the radio interface array
RadioIfaces	• Radio interface array. This is a structure of array containing the elements below. uint8_t RadioIfaces – See qaGobiApiTableRadioInterfaces.h for Radio Interfaces
Tlvresult	• Unpack Result

8.476.2 Field Documentation

8.476.2.1 uint32_t unpack_dms_GetDeviceCap_t::DataServiceCapability

8.476.2.2 uint32_t unpack_dms_GetDeviceCap_t::MaxRXChannelRate

8.476.2.3 uint32_t unpack_dms_GetDeviceCap_t::MaxTXChannelRate

8.476.2.4 uint8_t unpack_dms_GetDeviceCap_t::RadioIfaces[64]

8.476.2.5 uint32_t unpack_dms_GetDeviceCap_t::RadioIfacesSize

8.476.2.6 uint32_t unpack_dms_GetDeviceCap_t::SimCapability

8.476.2.7 uint16_t unpack_dms_GetDeviceCap_t::Tlvresult

8.477 unpack_dms_GetDeviceCapabilities_t Struct Reference

Data Fields

- uint32_t maxTxChannelRate
- uint32_t maxRxChannelRate
- uint32_t dataServiceCaCapability
- uint32_t simCapability
- uint32_t radioIfacesSize
- uint8_t RadioIfaces [255]

8.477.1 Detailed Description

Gets the device capabilities structure

Parameters

maxTxChannelRate	• Maximum transmission rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies
maxRxChannelRate	• Maximum reception rate (in bps) supported by the device • In multi-technology devices, this value will be the greatest rate among all supported technologies
dataServiceCaCapability	• CS/PS data service capability – 0 - No data services supported – 1 - Only Circuit Switched (CS) services supported – 2 - Only Packet Switched (PS) services supported – 3 - Simultaneous CS and PS – 4 - Non-simultaneous CS and PS
simCapability	• Device SIM capability – 0 - SIM not supported – 1 - SIM supported
radioIfacesSize	• Upon input, the maximum number of elements that the radio interface array can contain • Upon successful output, actual number of elements in the radio interface array
RadioIfaces[OUT]	• Radio interface array. This is a structure of array containing the elements below. uint8_t RadioIfaces – See qaGobiApiTableRadioInterfaces.h for Radio Interfaces

8.477.2 Field Documentation

8.477.2.1 uint32_t unpack_dms_GetDeviceCapabilities_t::dataServiceCaCapability

8.477.2.2 uint32_t unpack_dms_GetDeviceCapabilities_t::maxRxChannelRate

8.477.2.3 uint32_t unpack_dms_GetDeviceCapabilities_t::maxTxChannelRate

8.477.2.4 uint8_t unpack_dms_GetDeviceCapabilities_t::RadioIfaces[255]

8.477.2.5 uint32_t unpack_dms_GetDeviceCapabilities_t::radioIfacesSize

8.477.2.6 uint32_t unpack_dms_GetDeviceCapabilities_t::simCapability

8.478 unpack_dms_GetDeviceHardwareRev_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t Tlvresult

8.478.1 Detailed Description

Structure to store the hardware revision of the device

Parameters

stringSize	• The maximum number of characters (including NULL terminator) that the string array can contain
String	• NULL terminated Hardware Revision string • Maximum Length is 255 Bytes
Tlvresult	• Unpack Result

8.478.2 Field Documentation

8.478.2.1 char unpack_dms_GetDeviceHardwareRev_t::String[255]

8.478.2.2 uint8_t unpack_dms_GetDeviceHardwareRev_t::stringSize

8.478.2.3 uint16_t unpack_dms_GetDeviceHardwareRev_t::Tlvresult

8.479 unpack_dms_GetDeviceMfr_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t Tlvresult

8.479.1 Detailed Description

This structure is used to store unpack_dms_GetDeviceMfr parameters

Parameters

stringSize	• The maximum number of characters (including NULL terminator) that the string array can contain
String	• NULL terminated Device Manufacture string • Maximum Length is 255 Bytes
Tlvresult	• unpack Tlv result

8.479.2 Field Documentation

8.479.2.1 char unpack_dms_GetDeviceMfr_t::String[255]

8.479.2.2 uint8_t unpack_dms_GetDeviceMfr_t::stringSize

8.479.2.3 uint16_t unpack_dms_GetDeviceMfr_t::Tlvresult

8.480 unpack_dms_GetDeviceSerialNumbers_t Struct Reference

Data Fields

- uint8_t esnSize
- char ESNString [255]
- uint8_t imeiSize
- char IMEIString [255]
- uint8_t meidSize
- char MEIDString [255]
- uint8_t imeiSvnSize
- char ImeiSvnString [255]
- uint16_t Tlvresult

8.480.1 Detailed Description

This structure used to store unpack_dms_GetDeviceSerialNumbers parameters

Parameters

esnSize	• The maximum number of characters (including NULL terminator) that the ESN string array can contain
ESNString	• NULL-terminated ESN string. Empty string is returned when ESN is not supported/programmed • Maximum Length is 255 Bytes
imeiSize	• The maximum number of characters (including NULL terminator) that the IMEI string array can contain
IMEIString	• NULL terminated IMEI string. Empty string is returned when IMEI is not supported/programmed • Maximum Length is 255 Bytes
meidSize	• The maximum number of characters (including NULL terminator) that the MEID string array can contain
MEIDString	• NULL-terminated MEID string. Empty string is returned when MEID is not supported/programmed • Maximum Length is 255 Bytes
imeiSvnSize	• The maximum number of characters (including NULL terminator) that the IMEI SVN string array can contain.

ImeiSvnString	• NULL-terminated IMEI SVN string. Empty string is returned when IMEI SVN is not supported/programmed. • Maximum Length is 255 Bytes
Tlvresult	• unpack Tlv Result.

8.480.2 Field Documentation

8.480.2.1 uint8_t unpack_dms_GetDeviceSerialNumbers_t::esnSize

8.480.2.2 char unpack_dms_GetDeviceSerialNumbers_t::ESNString[255]

8.480.2.3 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSize

8.480.2.4 char unpack_dms_GetDeviceSerialNumbers_t::IMEIString[255]

8.480.2.5 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSvnSize

8.480.2.6 char unpack_dms_GetDeviceSerialNumbers_t::ImeiSvnString[255]

8.480.2.7 uint8_t unpack_dms_GetDeviceSerialNumbers_t::meidSize

8.480.2.8 char unpack_dms_GetDeviceSerialNumbers_t::MEIDString[255]

8.480.2.9 uint16_t unpack_dms_GetDeviceSerialNumbers_t::Tlvresult

8.481 unpack_dms_GetFirmwareInfo_t Struct Reference

Data Fields

- char modelid_str [20]
- char bootversion_str [85]
- char appversion_str [85]
- char sku_str [15]
- char packageid_str [85]
- char carrier_str [20]
- char priversion_str [16]
- char cur_carr_name [17]
- char cur_carr_rev [13]
- uint16_t Tlvresult

8.481.1 Detailed Description

This structure used to store unpack_dms_GetFirmwareInfo parameters

Parameters

modelid_str	- Mode ID String. • NULL-terminated Mode ID String.
bootversion_str	- Boot Version. • NULL-terminated Boot Version String.
appversion_str	- Application Version String. • NULL-terminated Application Version String.
sku_str	- SKU String. • NULL-terminated SKU String.
packageid_str	- Package ID String. • NULL-terminated Package ID String. • deprecated on EM/MC74xx(9x30) devices
carrier_str	- Carrier String. • NULL-terminated Carrier String.
priversion_str	- PRI Version String. • NULL-terminated PRI Version String.
cur_carr_name	- Current Carrier Name String. • NULL-terminated Current Carrier Name String.
cur_carr_rev	- Current Carrier Revision String. • NULL-terminated Current Carrier Revision String.
Tlvresult	- unpack Tlv Result.

8.481.2 Field Documentation

8.481.2.1 char unpack_dms_GetFirmwareInfo_t::appversion_str[85]

8.481.2.2 char unpack_dms_GetFirmwareInfo_t::bootversion_str[85]

8.481.2.3 char unpack_dms_GetFirmwareInfo_t::carrier_str[20]

8.481.2.4 char unpack_dms_GetFirmwareInfo_t::cur_carr_name[17]

8.481.2.5 char unpack_dms_GetFirmwareInfo_t::cur_carr_rev[13]

8.481.2.6 char unpack_dms_GetFirmwareInfo_t::modelid_str[20]

8.481.2.7 char unpack_dms_GetFirmwareInfo_t::packageid_str[85]

8.481.2.8 char unpack_dms_GetFirmwareInfo_t::priversion_str[16]

8.481.2.9 char unpack_dms_GetFirmwareInfo_t::sku_str[15]

8.481.2.10 uint16_t unpack_dms_GetFirmwareInfo_t::Tlvresult

8.482 unpack_dms_GetFirmwareRevision_t Struct Reference

Data Fields

- uint8_t amssSize
- char AMSSString [255]
- char PRIString [255]
- uint16_t Tlvresult

8.482.1 Detailed Description

This structure used to store unpack_dms_GetFirmwareRevision parameters

Parameters

amssSize	• Maximum number of characters (including NULL terminator) that the AMSS string array can contain.
AMSSString	• NULL-terminated AMSS revision string. • Maximum Length is 255 Bytes
PRIString[OUT]	• NULL-terminated PRI revision string. • Maximum Length is 255 Bytes
Tlvresult	• unpack Tlv Result.

8.482.2 Field Documentation

8.482.2.1 uint8_t unpack_dms_GetFirmwareRevision_t::amssSize

8.482.2.2 char unpack_dms_GetFirmwareRevision_t::AMSSString[255]

8.482.2.3 char unpack_dms_GetFirmwareRevision_t::PRIString[255]

8.482.2.4 uint16_t unpack_dms_GetFirmwareRevision_t::Tlvresult

8.483 unpack_dms_GetFirmwareRevisions_t Struct Reference

Data Fields

- uint8_t amssSize
- char AMSSString [255]
- uint8_t bootSize
- char BootString [255]
- uint8_t priSize
- char PRIString [255]
- uint16_t Tlvresult

8.483.1 Detailed Description

This structure used to store unpack_dms_GetFirmwareRevisions parameters

Parameters

amssSize	• Maximum number of characters (including NULL terminator) that the AMSS string array can contain
AMSSString	• NULL-terminated AMSS revision string • Maximum Length is 255 Bytes
bootSize	• Maximum number of characters (including NULL terminator) that the boot string array can contain
BootString	• NULL-terminated boot code revision string • Maximum Length is 255 Bytes
priSize	• Maximum number of characters (including NULL terminator) that the PRI string array can contain
PRIString	• NULL-terminated PRI revision string • Maximum Length is 255 Bytes
Tlvresult	• unpack Tlv Result

8.483.2 Field Documentation

8.483.2.1 uint8_t unpack_dms_GetFirmwareRevisions_t::amssSize

8.483.2.2 char unpack_dms_GetFirmwareRevisions_t::AMSSString[255]

8.483.2.3 uint8_t unpack_dms_GetFirmwareRevisions_t::bootSize

8.483.2.4 char unpack_dms_GetFirmwareRevisions_t::BootString[255]

8.483.2.5 uint8_t unpack_dms_GetFirmwareRevisions_t::priSize

8.483.2.6 char unpack_dms_GetFirmwareRevisions_t::PRIString[255]

8.483.2.7 uint16_t unpack_dms_GetFirmwareRevisions_t::Tlvresult

8.484 unpack_dms_GetFSN_t Struct Reference

Data Fields

- char String [255]

- uint16_t Tlvresult

8.484.1 Detailed Description

This structure used to store Factory Sequence Number parameter

Parameters

String	• Facorty Sequence Number • Maximum Length is 255 Bytes
Tlvresult	• Unpack Result

8.484.2 Field Documentation

8.484.2.1 char unpack_dms_GetFSN_t::String[255]

8.484.2.2 uint16_t unpack_dms_GetFSN_t::Tlvresult

8.485 unpack_dms_GetHardwareRevision_t Struct Reference

Data Fields

- char hwVer [255]

8.485.1 Detailed Description

This structure used to store unpack_dms_GetHardwareRevision parameters

Parameters

hwVer	- Hardware version • NULL-terminated string • Maximum Length is 255 Bytes

8.485.2 Field Documentation

8.485.2.1 char unpack_dms_GetHardwareRevision_t::hwVer[255]

8.486 unpack_dms_GetIMSI_t Struct Reference

Data Fields

- char imsi [255]
- uint16_t Tlvresult

8.486.1 Detailed Description

This structure used to store unpack_dms_GetIMSI parameters

Parameters

imsi	- IMSI no. • NULL-terminated String.
Tlvresult	- unpack Tlv Result.

8.486.2 Field Documentation

8.486.2.1 char unpack_dms_GetIMSI_t::imsi[255]

8.486.2.2 uint16_t unpack_dms_GetIMSI_t::Tlvresult

8.487 unpack_dms_GetManufacturer_t Struct Reference

Data Fields

- char manufacturer [255]
- uint16_t Tlvresult

8.487.1 Detailed Description

This structure is used to store device manufacturer information.

Parameters

manufacturer[OUT]	• NULL terminated string
Tlvresult	• Unpack Result

8.487.2 Field Documentation

8.487.2.1 char unpack_dms_GetManufacturer_t::manufacturer[255]

8.487.2.2 uint16_t unpack_dms_GetManufacturer_t::Tlvresult

8.488 unpack_dms_GetModelID_t Struct Reference

Data Fields

- char modelid [255]
- uint16_t Tlvresult

8.488.1 Detailed Description

This structure used to store unpack_dms_GetModelID parameters

Parameters

modelid	- Device model id. • NULL-terminated String.
Tlvresult	- unpack Tlv Result.

8.488.2 Field Documentation

8.488.2.1 char unpack_dms_GetModelID_t::modelid[255]

8.488.2.2 uint16_t unpack_dms_GetModelID_t::Tlvresult

8.489 unpack_dms_GetNetworkTime_t Struct Reference

Data Fields

- uint16_t source
- uint64_t timestamp
- uint16_t Tlvresult

8.489.1 Detailed Description

This structure used to store unpack_dms_GetNetworkTime parameters

Parameters

source	• Source of timestamp 0 - 32 kHz device clock 1 - CDMA network 2 - cdma2000 1xEV-DO network
timestamp	• Count of 1.25 ms that have elapsed from the start of GPS time (Jan 6, 1980)
Tlvresult	• Unpack Result

Note

The source of the timestamp provided specifies how the timestamp was determined. The first network time that is available will be returned. If no network time is available, the timestamp is taken from the 32 kHz slow-clock of the device.

8.489.2 Field Documentation

8.489.2.1 uint16_t unpack_dms_GetNetworkTime_t::source

8.489.2.2 uint64_t unpack_dms_GetNetworkTime_t::timestamp

8.489.2.3 uint16_t unpack_dms_GetNetworkTime_t::Tlvresult

8.490 unpack_dms_GetOfflineReason_t Struct Reference

Data Fields

- uint32_t * pReasonMask
- uint32_t * pbPlatform
- uint16_t Tlvresult

8.490.1 Detailed Description

This structure is used to store reason why the operating mode of the device is currently offline.

Parameters

pReasonMask[-OUT]	• Optional parameter • Bitmask of offline reasons – 0x00000001 - Host image configuration issue – 0x00000002 - PRI image configuration issue – 0x00000004 - PRI version incompatible – 0x00000008 - PRI copy issue – All others - Reserved
pbPlatform[OUT]	• Optional parameter • Is the device offline due to a platform restriction? – 0 - No – 1 - Yes
Tlvresult	• Unpack Result

8.490.2 Field Documentation

8.490.2.1 uint32_t* unpack_dms_GetOfflineReason_t::pbPlatform

8.490.2.2 uint32_t* unpack_dms_GetOfflineReason_t::pReasonMask

8.490.2.3 uint16_t unpack_dms_GetOfflineReason_t::Tlvresult

8.491 unpack_dms_GetPower_t Struct Reference

Data Fields

- uint32_t OperationMode
- uint32_t OfflineReason
- uint32_t HardwareControlledMode
- uint16_t Tlvresult

8.491.1 Detailed Description

This structure used to store unpack_dms_GetPower parameters

Parameters

OperationMode	- operating mode. • 0 - Online (default). • 1 - Low power (airplane) mode. • 2 - Factory test mode. • 3 - Offline. • 4 - Reset. • 5 - Power off. • 6 - Persistent low power (airplane) mode. • 7 - Mode - only low power.

Note

Valid transitions for Power Modes

- Online to Low Power, Persistent low power, Factory test, Offline or Shut Down
- Low power to online, Persistent low power, Offline, or Shut Down
- Persistent low power to Online, Low power, Offline or Shut down
- Factory test to online
- Offline to Reset

Parameters

OfflineReason	- offline reason.
Hardware-ControlledMode	- hardware restricted mode.
Tlvresult	- unpack Tlv Result.

8.491.2 Field Documentation

8.491.2.1 uint32_t unpack_dms_GetPower_t::HardwareControlledMode

8.491.2.2 uint32_t unpack_dms_GetPower_t::OfflineReason

8.491.2.3 uint32_t unpack_dms_GetPower_t::OperationMode

8.491.2.4 uint16_t unpack_dms_GetPower_t::Tlvresult

8.492 unpack_dms_GetPRLVersion_t Struct Reference

Data Fields

- uint8_t u8PRLPreference
- uint16_t u16PRLVersion
- uint16_t Tlvresult

8.492.1 Detailed Description

This structure used to store unpack_dms_GetPRLVersion parameters

Parameters

u8PRL-Preference	• PRL Preference – 0 - Unset – 1 - Set
u16PRLVersion	• PRL version of device.
Tlvresult	• unpack Tlv Result

8.492.2 Field Documentation

8.492.2.1 uint16_t unpack_dms_GetPRLVersion_t::Tlvresult

8.492.2.2 uint16_t unpack_dms_GetPRLVersion_t::u16PRLVersion

8.492.2.3 uint8_t unpack_dms_GetPRLVersion_t::u8PRLPreference

8.493 unpack_dms_GetSerialNumbers_t Struct Reference

Data Fields

- char esn [255]
- char imei_no [255]
- char meid [255]
- char imeisv_svn [255]

8.493.1 Detailed Description

This structure used to store unpack_dms_GetSerialNumbers parameters

Parameters

esn	- Electronic Serial Number of the device • NULL-terminated ESN string. Empty string is returned when ESN is not supported/programmed
imei_no	- International Mobile Equipment Identity of the device. • NULL terminated IMEI string. Empty string is returned when IMEI is not supported/programmed
meid	- Mobile Equipment Identifier of the device. • NULL-terminated MEID string. Empty string is returned when MEID is not supported/programmed
imeisv_svn	- NULL-terminated IMEI SVN string. Empty string is returned when IMEI SVN is not supported/programmed.imei software version revision

8.493.2 Field Documentation

8.493.2.1 char unpack_dms_GetSerialNumbers_t::esn[255]

8.493.2.2 char unpack_dms_GetSerialNumbers_t::imei_no[255]

8.493.2.3 char unpack_dms_GetSerialNumbers_t::imeisv_svn[255]

8.493.2.4 char unpack_dms_GetSerialNumbers_t::meid[255]

8.494 unpack_dms_GetUSBComp_t Struct Reference

Data Fields

- uint8_t USBComp [255]
- uint8_t NumSupUSBComps
- uint8_t SupUSBComps
- uint16_t Tlvresult

8.494.1 Detailed Description

This structure is used to store retrieved USB Composition

Parameters

USBComp	• Current USB Composition(optional parameter) • Values: – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM
NumSupUSB-Comps	• Number of supported USB compositions in the parameter to follow • Range - 0-255

SupUSBComps	• Optional parameter • List of supported USB compositions(1 Byte each - Max 255) • Total length is defined by pNumSupUSBComps parameter • Values: – 0..5 - Reserved (non-QMI) – 6 - DM NMEA AT QMI – 7 - DM NMEA AT QMI1 QMI2 QMI3 – 8 - DM NMEA AT MBIM – 9 - MBIM – 10 - NMEA MBIM – 11 - DM MBIM – 12 - DM NMEA MBIM 13-22 are combined compositions. One is for Win8 MBIM interfaces, another is for legacy QMI interfaces – 13 - 6 for QMI, 8 for MBIM – 14 - 6 for QMI, 9 for MBIM – 15 - 6 for QMI, 10 for MBIM – 16 - 6 for QMI, 11 for MBIM – 17 - 6 for QMI, 12 for MBIM – 18 - 7 for QMI, 8 for MBIM – 19 - 7 for QMI, 9 for MBIM – 20 - 7 for QMI, 10 for MBIM – 21 - 7 for QMI, 11 for MBIM – 22 - 7 for QMI, 12 for MBIM
Tlvresult	• unpack Tlv Result.

8.494.2 Field Documentation

8.494.2.1 uint8_t unpack_dms_GetUSBComp_t::NumSupUSBComps

8.494.2.2 uint8_t unpack_dms_GetUSBComp_t::SupUSBComps

8.494.2.3 uint16_t unpack_dms_GetUSBComp_t::Tlvresult

8.494.2.4 uint8_t unpack_dms_GetUSBComp_t::USBComp[255]

8.495 unpack_dms_GetVoiceNumber_t Struct Reference

Data Fields

- uint8_t voiceNumberSize
- char VoiceNumber [255]
- uint8_t minSize
- char MIN [255]
- uint16_t Tlvresult

8.495.1 Detailed Description

Structure to store the voice number in use by the device

Parameters

voiceNumber-Size	• Maximum number of characters (including NULL terminator) that the voice number array can contain.
VoiceNumber	• Voice number string: MDN or MS ISDN • Maximum Length is 255 Bytes
minSize	• Maximum number of characters (including NULL terminator) that the MIN array can contain.
MIN	• Optional Parameter • MIN string: Empty string returned when MIN is not supported/ programmed. • Maximum Length is 255 Bytes
Tlvresult	• Unpack Result

8.495.2 Field Documentation

8.495.2.1 char unpack_dms_GetVoiceNumber_t::MIN[255]

8.495.2.2 uint8_t unpack_dms_GetVoiceNumber_t::minSize

8.495.2.3 uint16_t unpack_dms_GetVoiceNumber_t::Tlvresult

8.495.2.4 char unpack_dms_GetVoiceNumber_t::VoiceNumber[255]

8.495.2.5 uint8_t unpack_dms_GetVoiceNumber_t::voiceNumberSize

8.496 unpack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.496.1 Detailed Description

This structure contains reset to factory default unpack

Parameters

Tlvresult	• Unpack Result

8.496.2 Field Documentation

8.496.2.1 uint16_t unpack_dms_ResetToFactoryDefaults_t::Tlvresult

8.497 unpack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.497.1 Detailed Description

This structure is used to store Set Service Activation Status callback parameter unpack.

Parameters

Tlvresult	• Unpack Result

8.497.2 Field Documentation

8.497.2.1 uint16_t unpack_dms_SetActivationStatusCallback_t::Tlvresult

8.498 unpack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t notused

8.498.1 Detailed Description

This structure is used to store unpack_dms_SetCrashAction parameters

Parameters

notused	• unpack Tlv result

8.498.2 Field Documentation

8.498.2.1 uint8_t unpack_dms_SetCrashAction_t::notused

8.499 unpack_dms_SetCustFeature_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.499.1 Detailed Description

This structure is used to store unpack_dms_SetCustFeature parameters

Parameters

Tlvresult	• unpack Tlv Result.

8.499.2 Field Documentation

8.499.2.1 uint16_t unpack_dms_SetCustFeature_t::Tlvresult

8.500 unpack_dms_SetCustFeaturesV2_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.500.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

Tlvresult	• Unpack Result

8.500.2 Field Documentation

8.500.2.1 uint16_t unpack_dms_SetCustFeaturesV2_t::Tlvresult

8.501 unpack_dms_SetEventReport_ind_t Struct Reference

Data Fields

- dms_ActivationStatusTlv ActivationStatusTlv
- dms_OperatingModeTlv OperatingModeTlv
- uint16_t Tlvresult

8.501.1 Detailed Description

DMS Event Report indication structure

Parameters

ActivationStatus-Tlv	• See dms_ActivationStatusTlv
OperatingMode-Tlv	• See dms_OperatingModeTlv

Tlvresult	• Unpack Result

8.501.2 Field Documentation

8.501.2.1 dms_ActivationStatusTlv unpack_dms_SetEventReport_ind_t::ActivationStatusTlv

8.501.2.2 dms_OperatingModeTlv unpack_dms_SetEventReport_ind_t::OperatingModeTlv

8.501.2.3 uint16_t unpack_dms_SetEventReport_ind_t::Tlvresult

8.502 unpack_dms_SetEventReport_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.502.1 Detailed Description

This structure is used to store unpack_dms_SetEventReport parameters.

Parameters

Tlvresult	• unpack Tlv result

8.502.2 Field Documentation

8.502.2.1 uint16_t unpack_dms_SetEventReport_t::Tlvresult

8.503 unpack_dms_SetFirmwarePreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.503.1 Detailed Description

This structure is used to store unpack_dms_SetFirmwarePreference parameters

Parameters

Tlvresult	• unpack Tlv result

8.503.2 Field Documentation

8.503.2.1 uint16_t unpack_dms_SetFirmwarePreference_t::Tlvresult

8.504 unpack_dms_SetPower_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.504.1 Detailed Description

Structure to store unpack the operating mode of the device.

Parameters

Tlvresult	• unpack Tlv Result.

8.504.2 Field Documentation

8.504.2.1 uint16_t unpack_dms_SetPower_t::Tlvresult

8.505 unpack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.505.1 Detailed Description

This structure is used to store unpack USB composition information

Parameters

Tlvresult	• unpack Tlv Result.

8.505.2 Field Documentation

8.505.2.1 uint16_t unpack_dms_SetUSBComp_t::Tlvresult

8.506 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult

8.506.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

type[OUT]	• type of reset or power down, possible values listed below: – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down
source[OUT]	• entity which initiated the reset or power down, possible values listed below: – 0 - unknown – 1 - user requested (AT!RESET, AT!BOOTHOLD, FW/PRI download – including host-initiated image switching) – 2 - hardware switch (W_DISABLE) – 3 - temperature critical – 4 - voltage critical – 5 - configuration update (SIM-based image switching, RMA reset, NVUPs which request a reset) – 6 - LWM2M (Light Weight M2M client (internal process for LWM2M)) – 7 - OMA-DM – 8 - FOTA
Tlvresult[OUT]	• Unpack Result

8.506.2 Field Documentation

8.506.2.1 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::source

8.506.2.2 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::Tlvresult

8.506.2.3 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::type

8.507 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult

8.507.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

type[OUT]	• type of reset or power down, possible values listed below: – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down
source[OUT]	• entity which initiated the reset or power down, possible values listed below: – 0 - unknown – 1 - user requested (AT!RESET, AT!BOOTHOLD, FW/PRI download – including host-initiated image switching) – 2 - hardware switch (W_DISABLE) – 3 - temperature critical – 4 - voltage critical – 5 - configuration update (SIM-based image switching, RMA reset, NVUPs which request a reset) – 6 - LWM2M (Light Weight M2M client (internal process for LWM2M)) – 7 - OMA-DM – 8 - FOTA
Tlvresult	• Unpack Result

8.507.2 Field Documentation

8.507.2.1 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::source

8.507.2.2 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_t::Tlvresult

8.507.2.3 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::type

8.508 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.508.1 Detailed Description

This structure contains set registration state for different indication unpack

Parameters

Tlvresult	• Unpack Result

8.508.2 Field Documentation

8.508.2.1 uint16_t unpack_dms_SLQSDmsSwiIndicationRegister_t::TlvResult

8.509 unpack_dms_SLQSGetBandCapability_t Struct Reference

Data Fields

- uint64_t bandCapability
- int is_LteBandCapability_Available
- uint64_t LteBandCapability
- int is_TdsBandCapability_Available
- uint64_t TdsBandCapability

8.509.1 Detailed Description

This structure contains the Band Capabilities response.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

bandCapability[-OUT]	Bitmask of bands supported by the device • Bit 0 - Band class 0, A-system • Bit 1 - Band class 0, B-system • Bit 2 - Band class 1, all blocks • Bit 3 - Band class 2 • Bit 4 - Band class 3, A-system • Bit 5 - Band class 4, all blocks • Bit 6 - Band class 5, all blocks • Bit 7 - GSM DCS band (1800) • Bit 8 - GSM Extended GSM (E-GSM) band (900) • Bit 9 - GSM Primary GSM (P-GSM) band (900) • Bit 10 - Band class 6 • Bit 11 - Band class 7 • Bit 12 - Band class 8 • Bit 13 - Band class 9 • Bit 14 - Band class 10 • Bit 15 - Band class 11 • Bit 16 - GSM 450 band • Bit 17 - GSM 480 band • Bit 18 - GSM 750 band • Bit 19 - GSM 850 band • Bit 20 - GSM railways GSM band (900) • Bit 21 - GSM PCS band (1900) • Bit 22 - WCDMA (Europe, Japan, and China) 2100 band • Bit 23 - WCDMA US PCS 1900 band • Bit 24 - WCDMA (Europe and China) DCS 1800 band • Bit 25 - WCDMA US 1700 band • Bit 26 - WCDMA US 850 band • Bit 27 - WCDMA Japan 800 band • Bit 28 - Band class 12 • Bit 29 - Band class 14 • Bit 30 - Reserved • Bit 31 - Band class 15 • Bits 32 through 47 - Reserved • Bit 48 - WCDMA Europe 2600 band • Bit 49 - WCDMA Europe and Japan 900 band • Bit 50 - WCDMA Japan 1700 band • Bits 51 through 55 - Reserved • Bit 56 - Band class 16 • Bit 57 - Band class 17 • Bit 58 - Band class 18 • Bit 59 - Band class 19

LteBand-Capability[OUT]	Bitmask of LTE bands supported by the device • Bit 0 - LTE EUTRAN Band 1 UL:1920-1980; DL: 2110-2170 • Bit 1 - LTE EUTRAN Band 2 UL:1850-1910; DL: 1930-1990 • Bit 2 - LTE EUTRAN Band 3 UL:1710-1785; DL: 1805-1880 • Bit 3 - LTE EUTRAN Band 4 UL:1710-1755; DL: 2110-2155 • Bit 4 - LTE EUTRAN Band 5 UL: 824-849; DL: 869-894 • Bit 5 - LTE EUTRAN Band 6 UL: 830-840; DL: 875-885 • Bit 6 - LTE EUTRAN Band 7 UL:2500-2570; DL: 2620-2690 • Bit 7 - LTE EUTRAN Band 8 UL: 880-915; DL: 925-960 • Bit 8 - LTE EUTRAN Band 9 UL:1749.9-1784.9; DL: 1844.9-1879.9 • Bit 9 - LTE EUTRAN Band 10 UL:1710-1770; DL: 2110-2170 • Bit 10 - LTE EUTRAN Band 11 UL:1427.9-1452.9; DL: 1475.9-1500.9 • Bit 11 - LTE EUTRAN Band 12 UL:698-716; DL: 728-746 • Bit 12 - LTE EUTRAN Band 13 UL: 777-787; DL: 746-756 • Bit 13 - LTE EUTRAN Band 14 UL: 788-798; DL: 758-768 • Bits 14 and 15 - Reserved • Bit 16 - LTE EUTRAN Band 17 UL: 704-716; DL: 734-746 • Bit 17 - LTE EUTRAN Band 18 UL: 815-830; DL: 860-875 • Bit 18 - LTE EUTRAN Band 19 UL: 830-845; DL: 875-890 • Bit 19 - LTE EUTRAN Band 20 UL: 832-862; DL: 791-821 • Bit 20 - LTE EUTRAN Band 21 UL: 1447.9-1462.9; DL: 1495.9-1510.9 • Bit 21 - Reserved • Bit 22 - LTE EUTRAN Band 23 UL: 2000-2020; DL: 2180-2200 • Bit 23 - LTE EUTRAN Band 24 UL: 1626.5-1660.5; DL: 1525-1559 • Bit 24 - LTE EUTRAN Band 25 UL: 1850-1915; DL: 1930-1995 • Bit 25 - LTE EUTRAN Band 26 UL: 814-849; DL: 859-894 • Bit 26 - Reserved • Bit 27 - LTE EUTRAN Band 28 UL: 703-748; DL: 758-803 • Bit 28 - LTE EUTRAN Band 29 UL: 1850-1910 or 1710-1755; DL: 716-728 • Bits 29 through 31 - Reserved • Bit 32 - LTE EUTRAN Band 33 UL: 1900-1920; DL: 1900-1920 • Bit 33 - LTE EUTRAN Band 34 UL: 2010-2025; DL: 2010-2025 • Bit 34 - LTE EUTRAN Band 35 UL: 1850-1910; DL: 1850-1910 • Bit 35 - LTE EUTRAN Band 36 UL: 1930-1990; DL: 1930-1990 • Bit 36 - LTE EUTRAN Band 37 UL: 1910-1930; DL: 1910-1930 • Bit 37 - LTE EUTRAN Band 38 UL: 2570-2620; DL: 2570-2620 • Bit 38 - LTE EUTRAN Band 39 UL: 1880-1920; DL: 1880-1920 • Bit 39 - LTE EUTRAN Band 40 UL: 2300-2400; DL: 2300-2400 • Bit 40 - LTE EUTRAN Band 41 UL: 2496-2690; DL: 2496-2690 • Bit 41 - LTE EUTRAN Band 42 UL: 3400-3600; DL: 3400-3600 • Bit 42 - LTE EUTRAN Band 43 UL: 3600-3800; DL: 3600-3800 • Bits 43 through 64 - Reserved

TdsBand-Capability[OUT]	Bitmask of TDS bands supported by the device. • Bit 0 - TDS Band A 1900 to 1920 MHz, 2010 to 2020 MHz • Bit 1 - TDS Band B 1850 to 1910 MHz, 1930 to 1990 MHz • Bit 2 - TDS Band C 1910 to 1930 MHz • Bit 3 - TDS Band D 2570 to 2620 MHz • Bit 4 - TDS Band E 2300 to 2400 MHz • Bit 5 - TDS Band F 1880 to 1920 MHz

8.509.2 Field Documentation

8.509.2.1 uint64_t unpack_dms_SLQSGetBandCapability_t::bandCapability

8.509.2.2 int unpack_dms_SLQSGetBandCapability_t::is_LteBandCapability_Available

8.509.2.3 int unpack_dms_SLQSGetBandCapability_t::is_TdsBandCapability_Available

8.509.2.4 uint64_t unpack_dms_SLQSGetBandCapability_t::LteBandCapability

8.509.2.5 uint64_t unpack_dms_SLQSGetBandCapability_t::TdsBandCapability

8.510 unpack_dms_SLQSGetERIFile_t Struct Reference

Data Fields

- uint16_t Tlvresult
- eriDataparams eriFile

8.510.1 Detailed Description

This structure contains Get ERI file unpack

Parameters

Tlvresult	• Unpack Result
eriFile	• Pointer to structure ERIFileparams • See eriDataparams for more information

8.510.2 Field Documentation

8.510.2.1 eriDataparams unpack_dms_SLQSGetERIFile_t::eriFile

8.510.2.2 uint16_t unpack_dms_SLQSGetERIFile_t::Tlvresult

8.511 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference

Data Fields

- uint16_t TlvresultTlvresult

8.511.1 Detailed Description

This structure contains Clear Dying GASP unpack

Parameters

Tlvresult	• Unpack Result

8.511.2 Field Documentation

8.511.2.1 uint16_t unpack_dms_SLQSSwiClearDyingGaspStatistics_t::Tlvresult

8.512 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint16_t TlvresultTlvresult
- crashInfoParams crashInfoParam

8.512.1 Detailed Description

This structure contains SWI get crash information unpack information

Parameters

Tlvresult	• Unpack Result
crashInfoParam[-OUT]	• See crashInfoParams

8.512.2 Field Documentation

8.512.2.1 crashInfoParams unpack_dms_SLQSSwiGetCrashInfo_t::crashInfoParam

8.512.2.2 uint16_t unpack_dms_SLQSSwiGetCrashInfo_t::Tlvresult

8.513 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference

Data Fields

- packgetDyingGaspCfg * pGetDyingGaspCfg
- uint16_t TlvresultTlvresult

8.513.1 Detailed Description

This structure contains Get Dying GASP Config unpack

Parameters

pGetDyingGasp-Cfg	• See packgetDyingGaspCfg
Tlvresult	• Unpack Result

8.513.2 Field Documentation

8.513.2.1 packgetDyingGaspCfg∗ unpack_dms_SLQSSwiGetDyingGaspCfg_t::pGetDyingGaspCfg

8.513.2.2 uint16_t unpack_dms_SLQSSwiGetDyingGaspCfg_t::Tlvresult

8.514 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference

Data Fields

- packgetDyingGaspStatistics ∗ pGetDyingGaspStatistics
- uint16_t Tlvresult

8.514.1 Detailed Description

This structure contains Get Dying GASP Statistics unpack.

Parameters

pGetDyingGasp-Statistics	• See packgetDyingGaspStatistics
Tlvresult	• Unpack Result

8.514.2 Field Documentation

8.514.2.1 packgetDyingGaspStatistics∗ unpack_dms_SLQSSwiGetDyingGaspStatistics_t::pGetDyingGaspStatistics

8.514.2.2 uint16_t unpack_dms_SLQSSwiGetDyingGaspStatistics_t::Tlvresult

8.515 unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference

Data Fields

- uint8_t numEntries
- image_info_t ∗ pCurrImgInfo
- char priver [16]
- char pkgver [16]

- char fwvers [16]
- char carrier [16]

8.515.1 Detailed Description

This structure is used to store unpack_dms_SLQSSwiGetFirmwareCurr parameters

Parameters

numEntries[IN/-OUT]	• Number of entries in the image list to follow • The size of the list pCurrImgInfo must be specified when calling the API
pCurrImgInfo[O-UT]	• Currently Active Image List
priver[OUT]	• PRI version of the currently running firmware
pkgver[OUT]	• Package version of the currently running firmware
fwvers[OUT]	• firmware version of the currently running firmware
carrier[OUT]	• Carrier string of the currently running firmware

8.515.2 Field Documentation

8.515.2.1 char unpack_dms_SLQSSwiGetFirmwareCurr_t::carrier[16]

8.515.2.2 char unpack_dms_SLQSSwiGetFirmwareCurr_t::fwvers[16]

8.515.2.3 uint8_t unpack_dms_SLQSSwiGetFirmwareCurr_t::numEntries

8.515.2.4 image_info_t* unpack_dms_SLQSSwiGetFirmwareCurr_t::pCurrImgInfo

8.515.2.5 char unpack_dms_SLQSSwiGetFirmwareCurr_t::pkgver[16]

8.515.2.6 char unpack_dms_SLQSSwiGetFirmwareCurr_t::priver[16]

8.516 unpack_dms_SLQSSwiGetFwUpdateStatus_t Struct Reference

Data Fields

- uint32_t ResCode
- uint8_t imgType
- uint32_t refData
- uint8_t refString [15]
- uint8_t logString [255]
- uint16_t Tlvresult

8.516.1 Detailed Description

This structure is used to store Firmware Update Status

Parameters

ResCode	• FW Update Result Code • Values: – 0x00000001 - Successful – 0xFFFFFFFF - Unknown (due to power off reset after firmware update) – 0x100000nn - File update errors while nn will be the exact error number: * 00 - General error – 0x200000nn - NVUP update errors while nn will be the exact error number: * 00 - General error – 0x40000nnn - FOTA update agent errors while nnn will be the exact error number: * 000 ∼ 0FF - Insignia defined error code * 100 ∼ 1FF - Sierra defined error code * See qaGobiApiTableFwDldErrorCodes.h for more detailed information – 0x800000nn - FDT/SSDP reported errors while nn will be the exact error number * See qaGobiApiTableFwDldErrorCodes.h for more detailed information
imgType	• Optional parameter • Firmware image type that failed the update
refData	• Optional parameter • Failed image reference data • This is normally the offset of the image that caused the failure
refString	• Optional parameter • Failed image reference string. This is normally the partition name of the image that caused the failure if applicable.
logString	• Optional parameter • Failed image reference string. This is normally the partition name of the image that caused the failure if applicable.
Tlvresult	• Unpack Result

8.516.2 Field Documentation

8.516.2.1 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::imgType

8.516.2.2 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::logString[255]

8.516.2.3 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refData

8.516.2.4 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refString[15]

8.516.2.5 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::ResCode

8.516.2.6 uint16_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::TlvResult

8.517 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference

Data Fields

- uint16_t TlvResult
- char manString [255]
- char modelString [255]
- char swVerString [255]
- char plasmaIDString [255]

8.517.1 Detailed Description

This structure contains SWI get host device info unpack information

Parameters

TlvResult	• unpack Result
manString[OUT]	• Host Device Manufacturer Name • Null terminated ASCII String
modelString[OUT]	• Host Device Model String • Null terminated ASCII String
swVerString[OUT]	• Host Device Software Version String • Null terminated ASCII string
plasmaIDString[OUT]	• Host Device Plasma ID String • Null terminated alphanumeric ASCII String.

8.517.2 Field Documentation

8.517.2.1 char unpack_dms_SLQSSwiGetHostDevInfo_t::manString[255]

8.517.2.2 char unpack_dms_SLQSSwiGetHostDevInfo_t::modelString[255]

8.517.2.3 char unpack_dms_SLQSSwiGetHostDevInfo_t::plasmaIDString[255]

8.517.2.4 char unpack_dms_SLQSSwiGetHostDevInfo_t::swVerString[255]

8.517.2.5 uint16_t unpack_dms_SLQSSwiGetHostDevInfo_t::TlvResult

8.518 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference

Data Fields

- uint16_t TlvResult
- char nameString [255]
- char versionString [255]

8.518.1 Detailed Description

This structure contains SWI get host os info unpack information

Parameters

TlvResult	• Unpack Result
nameString[OUT]	• Host operating system name
VersionString[OUT]	• Host operating system version

8.518.2 Field Documentation

8.518.2.1 char unpack_dms_SLQSSwiGetOSInfo_t::nameString[255]

8.518.2.2 uint16_t unpack_dms_SLQSSwiGetOSInfo_t::TlvResult

8.518.2.3 char unpack_dms_SLQSSwiGetOSInfo_t::versionString[255]

8.519 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference

Data Fields

- uint16_t TlvResult
- char meidString [8]

8.519.1 Detailed Description

This structure contains SWI get device serial number extension unpack information

Parameters

TlvResult	• pack Result
meidString[OUT]	• optional parameter, mobile equipment identifier

8.519.2 Field Documentation

8.519.2.1 char unpack_dms_SLQSSwiGetSerialNoExt_t::meidString[8]

8.519.2.2 uint16_t unpack_dms_SLQSSwiGetSerialNoExt_t::Tlvresult

8.520 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.520.1 Detailed Description

This structure contains set Dying GASP Config unpack

Parameters

Tlvresult	• Unpack Result

8.520.2 Field Documentation

8.520.2.1 uint16_t unpack_dms_SLQSSwiSetDyingGaspCfg_t::Tlvresult

8.521 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.521.1 Detailed Description

This structure contains SWI set host dev info unpack

Parameters

Tlvresult	• Unpack Result

8.521.2 Field Documentation

8.521.2.1 uint16_t unpack_dms_SLQSSwiSetHostDevInfo_t::Tlvresult

8.522 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.522.1 Detailed Description

This structure contains SWI set host OS info unpack

Parameters

Tlvresult	• Unpack Result

8.522.2 Field Documentation

8.522.2.1 uint16_t unpack_dms_SLQSSwiSetOSInfo_t::Tlvresult

8.523 unpack_dms_SLQSUIMGetState_t Struct Reference

Data Fields

- uint8_t state
- uint16_t Tlvresult

8.523.1 Detailed Description

This structure contains UIM get state unpack information

Parameters

state[OUT]	• UIM state
Tlvresult	• Unpack Result

8.523.2 Field Documentation

8.523.2.1 uint8_t unpack_dms_SLQSUIMGetState_t::state

8.523.2.2 uint16_t unpack_dms_SLQSUIMGetState_t::Tlvresult

8.524 unpack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facilityState
- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t Tlvresult

8.524.1 Detailed Description

This structure contains PIN retries status

Parameters

facilityState[OUT]	• Control key status – 0 - Deactivated – 1 - Activated – 2 - Blocked
verifyRetriesLeft[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. – 0xFFFFFFFF - Unknown
unblockRetriesLeft[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked i.e. UIM is unusable. – 0xFFFFFFFF - Unknown
Tlvresult	• Unpack Result

8.524.2 Field Documentation

8.524.2.1 uint8_t unpack_dms_UIMGetControlKeyStatus_t::facilityState

8.524.2.2 uint16_t unpack_dms_UIMGetControlKeyStatus_t::Tlvresult

8.524.2.3 uint8_t unpack_dms_UIMGetControlKeyStatus_t::unblockRetriesLeft

8.524.2.4 uint8_t unpack_dms_UIMGetControlKeyStatus_t::verifyRetriesLeft

8.525 unpack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint8_t stringSize
- uint8_t String [255]
- uint16_t Tlvresult

8.525.1 Detailed Description

This structure is used to store unpack_dms_UIMGetICCID parameters.

Parameters

stringSize	• Size of String.
String	• ICCID String. • Maximum Length is 255 Bytes

Tlvresult	• Unack result.

8.525.2 Field Documentation

8.525.2.1 uint8_t unpack_dms_UIMGetICCID_t::String[255]

8.525.2.2 uint8_t unpack_dms_UIMGetICCID_t::stringSize

8.525.2.3 uint16_t unpack_dms_UIMGetICCID_t::Tlvresult

8.526 unpack_dms_UIMGetPINStatus_t Struct Reference

Data Fields

- uint8_t p1Status
- uint8_t p1VerifyRetriesLeft
- uint8_t p1UnblockRetriesLeft
- uint8_t p2Status
- uint8_t p2VerifyRetriesLeft
- uint8_t p2UnblockRetriesLeft
- uint16_t Tlvresult

8.526.1 Detailed Description

This structure contains Get PIN Status unpack information

Parameters

p1Status[OUT]	• PIN1 status(0xFFFFFFFF - Unknown) – 0 - PIN not initialized – 1 - PIN enabled, not verified – 2 - PIN enabled, verified – 3 - PIN disabled – 4 - PIN blocked – 5 - PIN permanently blocked
p1VerifyRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left of PIN1, after which PIN will be blocked. – 0xFFFFFFFF - Unknown
p1Unblock-RetriesLeft[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left of PIN1, after which the PIN will be permanently blocked i.e. UIM is unusable. – 0xFFFFFFFF - Unknown

p2Status[OUT]	• PIN2 status(0xFFFFFFFF - Unknown) – 0 - PIN not initialized – 1 - PIN enabled, not verified – 2 - PIN enabled, verified – 3 - PIN disabled – 4 - PIN blocked – 5 - PIN permanently blocked
p2VerifyRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left of PIN2, after which PIN will be blocked. – 0xFFFFFFFF - Unknown
p2Unblock-RetriesLeft[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left of PIN2, after which the PIN will be permanently blocked i.e. UIM is unusable. – 0xFFFFFFFF - Unknown
Tlvresult	• Unpack Result

8.526.2 Field Documentation

8.526.2.1 uint8_t unpack_dms_UIMGetPINStatus_t::p1Status

8.526.2.2 uint8_t unpack_dms_UIMGetPINStatus_t::p1UnblockRetriesLeft

8.526.2.3 uint8_t unpack_dms_UIMGetPINStatus_t::p1VerifyRetriesLeft

8.526.2.4 uint8_t unpack_dms_UIMGetPINStatus_t::p2Status

8.526.2.5 uint8_t unpack_dms_UIMGetPINStatus_t::p2UnblockRetriesLeft

8.526.2.6 uint8_t unpack_dms_UIMGetPINStatus_t::p2VerifyRetriesLeft

8.526.2.7 uint16_t unpack_dms_UIMGetPINStatus_t::Tlvresult

8.527 unpack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint16_t Tlvresult

8.527.1 Detailed Description

This structure contains UIM Set control key protection unpack information

Parameters

verifyRetries-Left[OUT]	• Optional parameter • number of retries left after which the control key is blocked
Tlvresult	• Unpack Result

8.527.2 Field Documentation

8.527.2.1 uint16_t unpack_dms_UIMSetControlKeyProtection_t::Tlvresult

8.527.2.2 uint8_t unpack_dms_UIMSetControlKeyProtection_t::verifyRetriesLeft

8.528 unpack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t Tlvresult

8.528.1 Detailed Description

This structure contains PIN retries status

Parameters

verifyRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of retries left, after which PIN will be blocked. – 0xFFFFFFFF - Unknown
unblockRetries-Left[OUT]	• Optional parameter • Upon operational failure, this will indicate number of unblock retries left, after which the PIN will be permanently blocked i.e. UIM is unusable. – 0xFFFFFFFF - Unknown
Tlvresult[OUT]	• Unpack Result

8.528.2 Field Documentation

8.528.2.1 uint16_t unpack_dms_UIMSetPINProtection_t::Tlvresult

8.528.2.2 uint8_t unpack_dms_UIMSetPINProtection_t::unblockRetriesLeft

8.528.2.3 uint8_t unpack_dms_UIMSetPINProtection_t::verifyRetriesLeft

8.529 unpack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t unblockRetriesLeft
- uint16_t Tlvresult

8.529.1 Detailed Description

This structure contains UIM Set control key protection unpack information

Parameters

unblockRetries-Left[OUT]	• Optional parameter • number of unblock retries left after which the control key is permanently blocked
Tlvresult[OUT]	• Unpack Result

8.529.2 Field Documentation

8.529.2.1 uint16_t unpack_dms_UIMUnblockControlKey_t::Tlvresult

8.529.2.2 uint8_t unpack_dms_UIMUnblockControlKey_t::unblockRetriesLeft

8.530 unpack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageListSize
- FMSPrefImageList * pImageList
- uint16_t Tlvresult

8.530.1 Detailed Description

This structure contains the Get Image Preference information unpack

Parameters

listSize	• The number of elements in the image list
pListEntries	• Array of Image entries with size provided by previous field • See FMSImageElement
Tlvresult	• Unpack result

8.530.2 Field Documentation

8.530.2.1 uint32_t unpack_fms_GetImagesPreference_t::ImageListSize

8.530.2.2 FMSPrefImageList* unpack_fms_GetImagesPreference_t::pImageList

8.530.2.3 uint16_t unpack_fms_GetImagesPreference_t::Tlvresult

8.531 unpack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint32_t imagelistSize
- FMSImageList imageList
- uint16_t Tlvresult

8.531.1 Detailed Description

This structure contains the Get Stored Images unpack

Parameters

listSize	• The number of elements in the image list
imageList	• Array of Image entries with size provided by previous field • See FMSImageElement
Tlvresult	• Unpack result

8.531.2 Field Documentation

8.531.2.1 FMSImageList unpack_fms_GetStoredImages_t::imageList

8.531.2.2 uint32_t unpack_fms_GetStoredImages_t::imagelistSize

8.531.2.3 uint16_t unpack_fms_GetStoredImages_t::Tlvresult

8.532 unpack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageTypesSize
- uint8_t ImageTypes [255]
- uint16_t Tlvresult

8.532.1 Detailed Description

This structure contains the Set Images Preference unpack

Parameters

ImageTypesSize	• Image Type Size
ImageTypes	• Image Type
TIvresult	• Unpack result

8.532.2 Field Documentation

8.532.2.1 uint8_t unpack_fms_SetImagesPreference_t::ImageTypes[255]

8.532.2.2 uint32_t unpack_fms_SetImagesPreference_t::ImageTypesSize

8.532.2.3 uint16_t unpack_fms_SetImagesPreference_t::TIvresult

8.533 unpack_loc_BestAvailPos_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint32_t * pXid
- uint64_t * pLatitude
- uint64_t * pLongitude
- uint32_t * pHorUncCircular
- uint32_t * pAltitudeWrtEllipsoid
- uint32_t * pVertUnc
- uint64_t * pTimestampUtc
- uint32_t * pTimeUnc
- uint32_t * pHorUncEllipseSemiMinor
- uint32_t * pHorUncEllipseSemiMajor
- uint32_t * pHorUncEllipseOrientAzimuth
- uint8_t * pHorCirConf
- uint8_t * pHorEllpConf
- uint32_t * pHorReliability
- uint32_t * pSpeedHorizontal
- uint32_t * pSpeedUnc
- uint32_t * pAltitudeWrtMeanSeaLevel
- uint8_t * pVertConfidence
- uint32_t * pVertReliability
- uint32_t * pSpeedVertical
- uint32_t * pSpeedVerticalUnc
- uint32_t * pHeading
- uint32_t * pHeadingUnc
- uint32_t * pMagneticDeviation
- uint32_t * pTechnologyMask
- loc_precisionDilution * pPrecisionDilution
- loc_gpsTime * pGpsTime
- uint32_t * pTimeSrc
- loc_sensorDataUsage * pSensorDataUsage
- loc_svUsedforFix * pSvUsedforFix
- uint16_t TIvresult

8.533.1 Detailed Description

This structure contains Best Available Position

Parameters

status	• Valid values: – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
xid	Transaction ID that was specified in the Get Best Available Position request.
pLatitude	• Latitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude
pLongitude	• Longitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude
pHorUncCircular	• Horizontal position uncertainty. • Units - Meters
pAltitudeWrtEllipsoid	• Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883

pVertUnc	• Vertical uncertainty. • Units - Meters
pTimestampUtc	• UTC timestamp • Units - Milliseconds since Jan. 1, 1970
pTimeUnc	• Time uncertainty. • Units - Milliseconds
pHorUncEllipse-SemiMinor	• Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters
pHorUncEllipse-SemiMajor	• Semi-major axis of horizontal elliptical uncertainty. • Units: Meters
pHorUncEllipse-OrientAzimuth	• Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180
pHorCirConf	• Horizontal circular uncertainty confidence • Units: Precent • Range: 0 to 99
pHorEllpConf	• Horizontal elliptical uncertainty confidence • Units: Precent • Range: 0 to 99
pHorReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
pSpeed-Horizontal	• Horizontal speed. • Units - Meters/second
pSpeedUnc	• 3-D Speed uncertainty. • Units - Meters/second.

pAltitudeWrtMeanSeaLevel	• Altitude With Respect to Sea Level. • Units - Meters
pVertConfidence	• Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99
pVertReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
pSpeedVertical	• Vertical speed. • Units - Meters/second
pSpeedVerticalUnc	• Veritical speed • Units: Meters/second
pHeading	• Heading. • Units - Degree • Range 0 to 359.999
pHeadingUnc	• Heading uncertainty. • Units - Degree • Range 0 to 359.999
pMagneticDeviation	• Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north.
pTechnologyMask	• Values – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix

-pPrecision-Dilution	• See loc_precisionDilution for more information
-pGpsTime	• See loc_gpsTime for more information
pTimeSrc	• Values – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites.
-pSensorData-Usage	• See loc_sensorDataUsage for more information
-pSvUsedforFix	• See loc_svUsedforFix for more information

8.533.2 Field Documentation

8.533.2.1 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtEllipsoid

8.533.2.2 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtMeanSeaLevel

8.533.2.3 loc_gpsTime* unpack_loc_BestAvailPos_Ind_t::pGpsTime

8.533.2.4 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeading

8.533.2.5 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeadingUnc

8.533.2.6 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorCirConf

8.533.2.7 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorEllpConf

8.533.2.8 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorReliability

8.533.2.9 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncCircular

8.533.2.10 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseOrientAzimuth

8.533.2.11 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMajor

8.533.2.12 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMinor

8.533.2.13 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLatitude

8.533.2.14 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLongitude

8.533.2.15 uint32_t* unpack_loc_BestAvailPos_Ind_t::pMagneticDeviation

8.533.2.16 loc_precisionDilution* unpack_loc_BestAvailPos_Ind_t::pPrecisionDilution

8.533.2.17 loc_sensorDataUsage* unpack_loc_BestAvailPos_Ind_t::pSensorDataUsage

8.533.2.18 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedHorizontal

8.533.2.19 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedUnc

8.533.2.20 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVertical

8.533.2.21 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVerticalUnc

8.533.2.22 loc_svUsedforFix* unpack_loc_BestAvailPos_Ind_t::pSvUsedforFix

8.533.2.23 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTechnologyMask

8.533.2.24 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeSrc

8.533.2.25 uint64_t* unpack_loc_BestAvailPos_Ind_t::pTimestampUtc

8.533.2.26 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeUnc

8.533.2.27 uint8_t* unpack_loc_BestAvailPos_Ind_t::pVertConfidence

8.533.2.28 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertReliability

8.533.2.29 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertUnc

8.533.2.30 uint32_t* unpack_loc_BestAvailPos_Ind_t::pXid

8.533.2.31 uint32_t unpack_loc_BestAvailPos_Ind_t::status

8.533.2.32 uint16_t unpack_loc_BestAvailPos_Ind_t::TlvResult

8.534 unpack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- uint16_t TlvResult

8.534.1 Detailed Description

This structure contains LOC delete assist data unpack

Parameters

TlvResult	• Unpack result.

8.534.2 Field Documentation

8.534.2.1 uint16_t unpack_loc_Delete_Assist_Data_t::TlvResult

8.535 unpack_loc_DeleteAssistData_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t TlvResult

8.535.1 Detailed Description

Contain the parameters passed for SetLocDeleteAssistDataCallback by the device.

Parameters

status	• Status of the Delete Assist Data request • Valid values: – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
TlvResult	• unpack result

8.535.2 Field Documentation

8.535.2.1 uint32_t unpack_loc_DeleteAssistData_Ind_t::status

8.535.2.2 uint16_t unpack_loc_DeleteAssistData_Ind_t::Tlvresult

8.536 unpack_loc_EngineState_Ind_t Struct Reference

Data Fields

- uint32_t engineState
- uint16_t Tlvresult

8.536.1 Detailed Description

This structure contains LOC Engine State field.

Parameters

engineState	• Location engine state. • Valid values – 1 - Location engine is on – 2 - Loction engine is off
Tlvresult	• unpack result

8.536.2 Field Documentation

8.536.2.1 uint32_t unpack_loc_EngineState_Ind_t::engineState

8.536.2.2 uint16_t unpack_loc_EngineState_Ind_t::Tlvresult

8.537 unpack_loc_EventNMEA_Ind_t Struct Reference

Data Fields

- char NMEAData [201]
- uint16_t Tlvresult

8.537.1 Detailed Description

Contain the parameters passed for SetLocEventMaskNMEACallback by the device.

Parameters

NMEAData	• NMEA string • Type - NULL terminated string • Maximum string length (including NULL terminator) - 201

Tlvresult	• unpack result

8.537.2 Field Documentation

8.537.2.1 char unpack_loc_EventNMEA_Ind_t::NMEAData[201]

8.537.2.2 uint16_t unpack_loc_EventNMEA_Ind_t::Tlvresult

8.538 unpack_loc_EventRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.538.1 Detailed Description

This structure contains Event Register unpack

Parameters

Tlvresult	• Unpack result.

8.538.2 Field Documentation

8.538.2.1 uint16_t unpack_loc_EventRegister_t::Tlvresult

8.539 unpack_loc_GetServer_Ind_t Struct Reference

Data Fields

- uint32_t serverStatus
- uint32_t serverType
- loc_IPv4Info * pIPv4AddrInfo
- loc_IPv6Info * pIPv6AddrInfo
- loc_urlAddr * pURL
- uint16_t Tlvresult

8.539.1 Detailed Description

Contain the parameters passed for SetLocGetServerCallback by the device.

Parameters

serverStatus	• Status of the Get Server request. • Valid values – eQMI_LOC_SUCCESS (0) - Request was completed successfully – eQMI_LOC_GENERAL_FAILURE (1) - Request failed because of a general failure – eQMI_LOC_UNSUPPORTED (2) - Request failed because it is not supported – eQMI_LOC_INVALID_PARAMETER (3) - Request failed because it contained invalid parameters – eQMI_LOC_ENGINE_BUSY (4) - Request failed because the engine is busy – eQMI_LOC_PHONE_OFFLINE (5) - Request failed because the phone is offline – eQMI_LOC_TIMEOUT (6) - Request failed because it timed out – eQMI_LOC_CONFIG_NOT_SUPPORTED (7) - Request failed because an undefined configuration was requested – eQMI_LOC_INSUFFICIENT_MEMORY (8) - Request failed because the engine could not allocate sufficient memory for the request – eQMI_LOC_MAX_GEOFENCE_PROGRAMMED (9) - Request failed because the maximum number of Geofences are already programmed – eQMI_LOC_XTRA_VERSION_CHECK_FAILURE (10) - Location service failed because of an XTRA version-based file format check failure
serverType	• Type of server. • Valid values – eQMI_LOC_SERVER_TYPE_CDMA_PDE (1) - Server type is CDMA PDE – eQMI_LOC_SERVER_TYPE_CDMA_MPC (2) - Server type is CDMA MPC – eQMI_LOC_SERVER_TYPE_UMTS_SLP (3) - Server type is UMTS SLP – eQMI_LOC_SERVER_TYPE_CUSTOM_PDE (4) - Server type is custom PDE
pIPv4AddrInfo	• See loc_IPv4Info for more information.
pIPv6AddrInfo	• See loc_IPv6Info for more information.
pURL	• See loc_urlAddr for more information.

8.539.2 Field Documentation

8.539.2.1 loc_IPv4Info* unpack_loc_GetServer_Ind_t::pIPv4AddrInfo

8.539.2.2 loc_IPv6Info* unpack_loc_GetServer_Ind_t::pIPv6AddrInfo

8.539.2.3 loc_urlAddr* unpack_loc_GetServer_Ind_t::pURL

8.539.2.4 uint32_t unpack_loc_GetServer_Ind_t::serverStatus

8.539.2.5 uint32_t unpack_loc_GetServer_Ind_t::serverType

8.539.2.6 uint16_t unpack_loc_GetServer_Ind_t::TlvResult

8.540 unpack_loc_GnssSvInfo_Ind_t Struct Reference

Data Fields

- uint8_t altitudeAssumed
- loc_satelliteInfo * pSatelliteInfo
- uint16_t TlvResult

8.540.1 Detailed Description

Contain the parameters passed for SetLocGnssSvInfoCallback by the device.

Parameters

altitudeAssumed	• Indicates whether altitude is assumed or calculated – 0x00 (FALSE) - Valid altitude is calculated – 0x01 (TRUE) - Valid altitude is assumed; there may not be enough satellites to determine precise altitude
pSatelliteInfo	• See loc_satelliteInfo for more information.
TlvResult	• unpack result

8.540.2 Field Documentation

8.540.2.1 uint8_t unpack_loc_GnssSvInfo_Ind_t::altitudeAssumed

8.540.2.2 loc_satelliteInfo* unpack_loc_GnssSvInfo_Ind_t::pSatelliteInfo

8.540.2.3 uint16_t unpack_loc_GnssSvInfo_Ind_t::TlvResult

8.541 unpack_loc_PositionRpt_Ind_t Struct Reference

Data Fields

- uint32_t sessionStatus
- uint8_t sessionId
- uint64_t * pLatitude
- uint64_t * pLongitude
- uint32_t * pHorUncCircular
- uint32_t * pHorUncEllipseSemiMinor
- uint32_t * pHorUncEllipseSemiMajor
- uint32_t * pHorUncEllipseOrientAzimuth
- uint8_t * pHorConfidence
- uint32_t * pHorReliability
- uint32_t * pSpeedHorizontal
- uint32_t * pSpeedUnc
- uint32_t * pAltitudeWrtEllipsoid
- uint32_t * pAltitudeWrtMeanSeaLevel

- uint32_t * pVertUnc
- uint8_t * pVertConfidence
- uint32_t * pVertReliability
- uint32_t * pSpeedVertical
- uint32_t * pHeading
- uint32_t * pHeadingUnc
- uint32_t * pMagneticDeviation
- uint32_t * pTechnologyMask
- loc_precisionDilution * pPrecisionDilution
- uint64_t * pTimestampUtc
- uint8_t * pLeapSeconds
- loc_gpsTime * pGpsTime
- uint32_t * pTimeUnc
- uint32_t * pTimeSrc
- loc_sensorDataUsage * pSensorDataUsage
- uint32_t * pFixId
- loc_svUsedforFix * pSvUsedforFix
- uint8_t * pAltitudeAssumed
- uint16_t Tlvresult

8.541.1 Detailed Description

This structure contains Event Position Report Indication unpack

Parameters

sessionStatus	• Values – 0 - Session was successful – 1 - Session is still in progress; further position reports will be generated until either the fix criteria specified by the client are met or the client response timeout occurs. – 2 - Session failed.. – 3 - Fix request failed because the session timed out. – 4 - Fix request failed because the session was ended by the user. – 5 - Fix request failed due to bad parameters in the request. – 6 - Fix request failed because the phone is offline. – 7 - Fix request failed because the engine is locked
sessionId	• ID of the session that was specified in the Start request • Range - 0 to 255
pLatitude	• Latitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -90.0 to 90.0 • Positive values indicate northern latitude • Negative values indicate southern latitude

pLongitude	• Longitude (specified in WGS84 datum) • Type - Floating point • Units - Degrees • Range - -180.0 to 180.0 • Positive values indicate eastern latitude • Negative values indicate western latitude
pHorUncCircular	• Horizontal position uncertainty. • Units - Meters
pHorUncEllipse-SemiMinor	• Semi-minor axis of horizontal elliptical uncertainty. • Units - Meters
pHorUncEllipse-SemiMajor	• Semi-major axis of horizontal elliptical uncertainty. • Units: Meters
pHorUncEllipse-OrientAzimuth	• Elliptical horizontal uncertainty azimuth of orientation. • Units - Decimal degrees • Range - 0 to 180
pHorConfidence	• Horizontal uncertainty confidence. • If both elliptical and horizontal uncertainties are specified in this message, the confidence corresponds to the elliptical uncertainty. • Units - Percentage • Range 0-99
pHorReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk – 2 - Location reliability is low; little or no cross-checking is possible. – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
pSpeed-Horizontal	• Horizontal speed. • Units - Meters/second
pSpeedUnc	• 3-D Speed uncertainty. • Units - Meters/second.

pAltitudeWrt-Ellipsoid	• Altitude With Respect to WGS84 Ellipsoid. • Units - Meters • Range -500 to 15883
pAltitudeWrt-MeanSeaLevel	• Altitude With Respect to Sea Level. • Units - Meters
pVertUnc	• Vertical uncertainty. • Units - Meters
pVertConfidence	• Vertical uncertainty confidence. • Units - Percentage • Range 0 to 99
pVertReliability	• Values – 0 - Location reliability is not set. – 1 - Location reliability is very low; use it at your own risk. – 2 - Location reliability is low; little or no cross-checking is possible – 3 - Location reliability is medium; limited cross-check passed – 4 - Location reliability is high; strong cross-check passed
pSpeedVertical	• Vertical speed. • Units - Meters/second
pHeading	• Heading. • Units - Degree • Range 0 to 359.999
pHeadingUnc	• Heading uncertainty. • Units - Degree • Range 0 to 359.999
pMagnetic-Deviation	• Difference between the bearing to true north and the bearing shown on a magnetic compass. The deviation is positive when the magnetic north is east of true north.

pTechnology-Mask	• Values – 0x00000001 - Satellites were used to generate the fix – 0x00000002 - Cell towers were used to generate the fix – 0x00000004 - Wi-Fi access points were used to generate the fix – 0x00000008 - Sensors were used to generate the fix – 0x00000010 - Reference Location was used to generate the fix – 0x00000020 - Coarse position injected into the location engine was used to generate the fix – 0x00000040 - AFLT was used to generate the fix – 0x00000080 - GNSS and network-provided measurements were used to generate the fix
-pPrecision-Dilution	• See loc_precisionDilution for more information
pTimestampUtc	• UTC timestamp • Units - Milliseconds since Jan. 1, 1970
pLeapSeconds	• Leap second information. If leapSeconds is not available, timestampUtc is calculated based on a hard-coded value for leap seconds. • Units - Seconds
-pGpsTime	• See loc_gpsTime for more information
pTimeUnc	• Time uncertainty. • Units - Milliseconds

pTimeSrc	• Values – 0 - Invalid time. – 1 - Time is set by the 1X system. – 2 - Time is set by WCDMA/GSM time tagging. – 3 - Time is set by an external injection. – 4 - Time is set after decoding over-the-air GPS navigation data from one GPS satellite. – 5 - Time is set after decoding over-the-air GPS navigation data from multiple satellites. – 6 - Both time of the week and the GPS week number are known. – 7 - Time is set by the position engine after the fix is obtained – 8 - Time is set by the position engine after performing SFT, this is done when the clock time uncertainty is large. – 9 - Time is set after decoding GLO satellites. – 10- Time is set after transforming the GPS to GLO time – 11- Time is set by the sleep time tag provided by the WCDMA network. – 12- Time is set by the sleep time tag provided by the GSM network – 13- Source of the time is unknown – 14- Time is derived from the system clock (better known as the slow clock); GNSS time is maintained irrespective of the GNSS receiver state – 15- Time is set after decoding QZSS satellites. – 16- Time is set after decoding BDS satellites.
-pSensorData-Usage	• See loc_sensorDataUsage for more information
pFixId	• Fix count for the session. Starts with 0 and increments by one for each successive position report for a particular session.
-pSvUsedforFix	• See loc_svUsedforFix for more information
pAltitude-Assumed	• Indicates whether altitude is assumed or calculated.

- Value
  - 0x00 - Altitude is calculated
  - 0x01 - Altitude is assumed

8.541.2 Field Documentation

8.541.2.1 uint8_t* unpack_loc_PositionRpt_Ind_t::pAltitudeAssumed

8.541.2.2 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtEllipsoid

8.541.2.3 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtMeanSeaLevel

8.541.2.4 uint32_t* unpack_loc_PositionRpt_Ind_t::pFixId

8.541.2.5 loc_gpsTime* unpack_loc_PositionRpt_Ind_t::pGpsTime

8.541.2.6 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeading

8.541.2.7 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeadingUnc

8.541.2.8 uint8_t* unpack_loc_PositionRpt_Ind_t::pHorConfidence

8.541.2.9 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorReliability

8.541.2.10 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncCircular

8.541.2.11 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseOrientAzimuth

8.541.2.12 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMajor

8.541.2.13 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMinor

8.541.2.14 uint64_t* unpack_loc_PositionRpt_Ind_t::pLatitude

8.541.2.15 uint8_t* unpack_loc_PositionRpt_Ind_t::pLeapSeconds

8.541.2.16 uint64_t* unpack_loc_PositionRpt_Ind_t::pLongitude

8.541.2.17 uint32_t* unpack_loc_PositionRpt_Ind_t::pMagneticDeviation

8.541.2.18 loc_precisionDilution* unpack_loc_PositionRpt_Ind_t::pPrecisionDilution

8.541.2.19 loc_sensorDataUsage* unpack_loc_PositionRpt_Ind_t::pSensorDataUsage

8.541.2.20 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedHorizontal

8.541.2.21 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedUnc

8.541.2.22 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedVertical

8.541.2.23 loc_svUsedforFix* unpack_loc_PositionRpt_Ind_t::pSvUsedforFix

8.541.2.24 uint32_t* unpack_loc_PositionRpt_Ind_t::pTechnologyMask

8.541.2.25 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeSrc

8.541.2.26 uint64_t* unpack_loc_PositionRpt_Ind_t::pTimestampUtc

8.541.2.27 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeUnc

8.541.2.28 uint8_t* unpack_loc_PositionRpt_Ind_t::pVertConfidence

8.541.2.29 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertReliability

8.541.2.30 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertUnc

8.541.2.31 uint8_t unpack_loc_PositionRpt_Ind_t::sessionId

8.541.2.32 uint32_t unpack_loc_PositionRpt_Ind_t::sessionStatus

8.541.2.33 uint16_t unpack_loc_PositionRpt_Ind_t::Tlvresult

8.542 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult

8.542.1 Detailed Description

This structure contains LOC Set External Power Configure status field.

Parameters

status	• Valid values – 0 - Request was completed successfully – 1 - Request failed because of a general failure. – 2 - Request failed because it is not supported. – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - engine could not allocate sufficient memory – 9 - Request failed because the maximum number of Geofences are already programmed – 10 -Location service failed because of an XTRA version-based file format check failure
Tlvresult	• unpack result

8.542.2 Field Documentation

8.542.2.1 uint32_t unpack_loc_SetExtPowerConfig_Ind_t::status

8.542.2.2 uint16_t unpack_loc_SetExtPowerConfig_Ind_t::Tlvresult

8.543 unpack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.543.1 Detailed Description

This structure contains Set Ext Power State unpack

Parameters

Tlvresult	• Unpack result.

8.543.2 Field Documentation

8.543.2.1 uint16_t unpack_loc_SetExtPowerState_t::Tlvresult

8.544 unpack_loc_SetOperationMode_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult

8.544.1 Detailed Description

This structure contains LOC Set External Power Configure status field.

Parameters

status	• Status of the Set Operation Mode request. • Valid values: – 0 - Request was completed successfully – 1 - Request failed because of a general failure – 2 - Request failed because it is not supported – 3 - Request failed because it contained invalid parameters – 4 - Request failed because the engine is busy – 5 - Request failed because the phone is offline – 6 - Request failed because it timed out – 7 - Request failed because an undefined configuration was requested – 8 - Request failed because the engine could not allocate sufficient memory for the request – 9 - Request failed because the maximum number of Geofences are already programmed – 10 - Location service failed because of an XTRA version-based file format check failure
Tlvresult	• unpack result

8.544.2 Field Documentation

8.544.2.1 uint32_t unpack_loc_SetOperationMode_Ind_t::status

8.544.2.2 uint16_t unpack_loc_SetOperationMode_Ind_t::Tlvresult

8.545 unpack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.545.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

Tlvresult	• Unpack result.

8.545.2 Field Documentation

8.545.2.1 uint16_t unpack_loc_SetOperationMode_t::Tlvresult

8.546 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.546.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

Tlvresult	• Unpack result.

8.546.2 Field Documentation

8.546.2.1 uint16_t unpack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.547 unpack_loc_Start_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.547.1 Detailed Description

This structure contains Start LOC unpack

Parameters

Tlvresult	• Unpack result.

8.547.2 Field Documentation

8.547.2.1 uint16_t unpack_loc_Start_t::Tlvresult

8.548 unpack_loc_Stop_t Struct Reference

Data Fields

- uint16_t TlvResult

8.548.1 Detailed Description

This structure contains Stop LOC unpack

Parameters

TlvResult	• Unpack result.

8.548.2 Field Documentation

8.548.2.1 uint16_t unpack_loc_Stop_t::TlvResult

8.549 unpack_nas_GetCDMANetworkParameters_t Struct Reference

Data Fields

- uint8_t SCI
- uint8_t SCM
- uint8_t RegHomeSID
- uint8_t RegForeignSID
- uint8_t RegForeignNID
- uint8_t ForceRev0
- uint8_t CustomSCP
- uint32_t Protocol
- uint32_t Broadcast
- uint32_t Application
- uint32_t Roaming

8.549.1 Detailed Description

Gets the current CDMA network parameters

Parameters

SCI	• Slot cycle index – 0xFF-Unknown
SCM	• Station class mark – 0xFF-Unknown
RegHomeSID	• Register on home SID – 0 - Disabled – 1 - Enabled – 0xFF - Unknown

RegForeignSID	• Register on foreign SID – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
RegForeignNID	• Register on foreign NID – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
ForceRev0	• Force CDMA 1x-EV-DO Rev. 0 mode – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
CustomSCP	• Use a custom config for CDMA 1x-EV-DO SCP – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
Protocol	• Protocol mask for custom SCP config – 0x00000001 - Subtype 2 Physical Layer – 0x00000002 - Enhanced CCMAC – 0x00000004 - Enhanced ACMAC – 0x00000008 - Enhanced FTCMAC – 0x00000010 - Subtype 3 RTCMAC – 0x00000020 - Subsystem 1 RTCMAC – 0x00000040 - Enhanced Idle – 0x00000080 - Generic Multimode Capable Disc Port – 0xFFFFFFFF - Unknown
Broadcast	• Broadcast mask for custom SCP config – 0x00000001 - Generic broadcast enabled – 0xFFFFFFFF - Unknown
Application	• Application mask for custom SCP config – 0x00000001 - SN Multiflow Packet Application – 0x00000002 - SN Enhanced Multiflow Packet Application – 0xFFFFFFFF - Unknown

Roaming	• Roaming preference – 0 - Automatic – 1 - Home Only – 2 - Affiliated Roaming Only – 3 - Home and Affiliated Roaming – 0xFFFFFFFF - Unknown

8.549.2 Field Documentation

8.549.2.1 uint32_t unpack_nas_GetCDMANetworkParameters_t::Application

8.549.2.2 uint32_t unpack_nas_GetCDMANetworkParameters_t::Broadcast

8.549.2.3 uint8_t unpack_nas_GetCDMANetworkParameters_t::CustomSCP

8.549.2.4 uint8_t unpack_nas_GetCDMANetworkParameters_t::ForceRev0

8.549.2.5 uint32_t unpack_nas_GetCDMANetworkParameters_t::Protocol

8.549.2.6 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignNID

8.549.2.7 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignSID

8.549.2.8 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegHomeSID

8.549.2.9 uint32_t unpack_nas_GetCDMANetworkParameters_t::Roaming

8.549.2.10 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCI

8.549.2.11 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCM

8.550 unpack_nas_GetHomeNetwork3GPP2_t Struct Reference

Data Fields

- uint16_t * pMCC
- uint16_t * pMNC
- uint8_t nameSize
- uint8_t * pName
- uint16_t * pSID
- uint16_t * pNID
- uint16_t * pNw2MCC
- uint16_t * pNw2MNC
- uint8_t * pNw2DescDisp
- uint8_t * pNw2DescEnc
- uint8_t * pNw2DescLen
- uint8_t * pNw2Name

8.550.1 Detailed Description

Structure for HomeNetwork3GPP2 unpack.

Parameters

pMCC	• Mobile country code (UMTS only).
pMNC	• Mobile network code (UMTS only).
nameSize	• Maximum number of characters (including NULL terminator) that network name array can contain (UMTS only).
pName	• Network name or description represented as a NULL terminated string (empty string returned when unknown) (UMTS only).
pSID	• Home network system ID – 0xFFFF - Unknown. – Only applies to cdma2000
pNID	• Home network ID – 0xFFFF - Unknown. – Only applies to cdma2000
pNw2MCC	• Mobile country code (3GPP2 only). • Range : 0 to 999
pNw2MNC	• Mobile network code (3GPP2 only). • Range : 0 to 999
pNw2DescDisp	• Network Name Display (3GPP2 only). -Valid Value – 0x00 - Do not display – 0x01 - Display – 0xFF - Unknown
pNw2DescEnc	• Encoding of the network description (3GPP2 only). • Valid Value – 0x00 - Octet, unspecified – 0x02 - 7-bit ASCII (liteqmi_helper_decode7bitAsciiEncString) – 0x04 - Unicode – 0x09 - GSM 7-bit default
nw2DescLen	• Network Description Length (3GPP2 only).

pNw2Name	• Network Name (3GPP2 only).

8.550.2 Field Documentation

8.550.2.1 uint8_t unpack_nas_GetHomeNetwork3GPP2_t::nameSize

8.550.2.2 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMCC

8.550.2.3 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMNC

8.550.2.4 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pName

8.550.2.5 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNID

8.550.2.6 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescDisp

8.550.2.7 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescEnc

8.550.2.8 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescLen

8.550.2.9 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MCC

8.550.2.10 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MNC

8.550.2.11 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2Name

8.550.2.12 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pSID

8.551 unpack_nas_GetHomeNetwork_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- char name [255]
- uint16_t sid
- uint16_t nid

8.551.1 Detailed Description

Parameters

mcc	• Mobile country code (UMTS only).
mnc	• Mobile network code (UMTS only).
name	• Network name or description represented as a NULL terminated string (empty string returned when unknown) (UMTS only).

sid	• Home network system ID – 0xFFFF - Unknown. – Only applies to cdma2000
nid	• Home network ID – 0xFFFF - Unknown. – Only applies to cdma2000

8.551.2 Field Documentation

8.551.2.1 uint16_t unpack_nas_GetHomeNetwork_t::mcc

8.551.2.2 uint16_t unpack_nas_GetHomeNetwork_t::mnc

8.551.2.3 char unpack_nas_GetHomeNetwork_t::name[255]

8.551.2.4 uint16_t unpack_nas_GetHomeNetwork_t::nid

8.551.2.5 uint16_t unpack_nas_GetHomeNetwork_t::sid

8.552 unpack_nas_GetNetworkPreference_t Struct Reference

Data Fields

- uint32_t ActiveTechPref
- uint32_t Duration
- uint32_t PersistentTechPref
- uint16_t Tlvresult

8.552.1 Detailed Description

Returns the network registration preference. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use unpack_nas_SLQSGetSys-SelectionPref() for new firmware versions and new modules

Parameters

TechnologyPref[-OUT]	• Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved
Duration[OUT]	• Duration of active preference – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call
Persistent-TechnologyPref[-OUT]	• Bit field representing persistent radio technology preference – Same representation as the pTechnologyPref parameter
Tlvresult	• unpack result

8.552.2 Field Documentation

8.552.2.1 uint32_t unpack_nas_GetNetworkPreference_t::ActiveTechPref

8.552.2.2 uint32_t unpack_nas_GetNetworkPreference_t::Duration

8.552.2.3 uint32_t unpack_nas_GetNetworkPreference_t::PersistentTechPref

8.552.2.4 uint16_t unpack_nas_GetNetworkPreference_t::Tlvresult

8.553 unpack_nas_GetRFInfo_t Struct Reference

Data Fields

- uint8_t instancesSize
- RFBandInfoElements RFBandInfoElements [255]

8.553.1 Detailed Description

Structur to strore the RFInfoList

Parameters

instanceSize	• Upon input, maximum number of elements that the RF info instances array can contain. • Upon successful output, actual number of elements in RF info instances array.
RFBandInfo-Elements	• RF info instances array – See RFBandInfoElements for more information

8.553.2 Field Documentation

8.553.2.1 uint8_t unpack_nas_GetRFInfo_t::instancesSize

8.553.2.2 RFBandInfoElements unpack_nas_GetRFInfo_t::RFBandInfoElements[255]

8.554 unpack_nas_GetServingNetwork_t Struct Reference

Data Fields

- uint32_t RegistrationState
- uint32_t CSDomain
- uint32_t PSDomain
- uint32_t RAN
- uint8_t RadioIfacesSize
- uint8_t RadioIfaces [255]
- uint32_t Roaming
- uint16_t MCC
- uint16_t MNC
- uint8_t nameSize
- uint8_t Name [255]
- uint8_t DataCapsLen
- uint8_t DataCaps [255]

8.554.1 Detailed Description

Parameters

Registration-State	• Registration state: – 0 - Not registered – 1 - Registered – 2 - Searching/Not Registered – 3 - Registration Denied – 4 - Unknown
CSDomain	• Circuit switch domain status: – 0 - Unknown/Not Applicable – 1 - Attached – 2 - Detached

PSDomain	• Packet switch domain status – 0 - Unknown/Not Applicable – 1 - Attached – 2 - Detached
RAN	• Type of radio access network on which mobile is registered: – 0 - Unknown – 1 - cdma2000 network – 2 - UMTS network
RadioIfacesSize	• Upon input, maximum number of elements that the radio interface array contain. • Upon successful output, actual number of elements in the radio interface array.
RadioIfaces	• An array of Radio Interface Technology – See qaGobiApiTableRadioInterfaces.h for the Radio Interface Technologies
Roaming	• Roaming indicator
MCC	• Mobile country code
MNC	• Mobile network code
nameSize	• Maximum number of characters (including NULL terminator) that network name array can contain; applicable only for UMTS networks
Name	• Network name or description represented as a NULL terminated string; empty string is returned when unknown; applicable only for UMTS networks
DataCapsLen[IN/OUT]	• Upon input, the maximum number of elements the data capabilities array can contain. • Upon output, the actual number of elements in the data capabilities array.

DataCaps[OUT]	• Data capabilities array of unsigned long type – 1 - GPRS – 2 - EDGE – 3 - HSDPA – 4 - HSUPA – 5 - WCDMA – 6 - CDMA 1xRTT – 7 - CDMA 1xEV-DO Rev 0 – 8 - CDMA 1xEV-DO Rev. A – 9 - GSM – 10 - EVDO Rev. B – 11 - LTE – 12 - HSDPA Plus – 13 - Dual Carrier HSDPA Plus

8.554.2 Field Documentation

8.554.2.1 uint32_t unpack_nas_GetServingNetwork_t::CSDomain

8.554.2.2 uint8_t unpack_nas_GetServingNetwork_t::DataCaps[255]

8.554.2.3 uint8_t unpack_nas_GetServingNetwork_t::DataCapsLen

8.554.2.4 uint16_t unpack_nas_GetServingNetwork_t::MCC

8.554.2.5 uint16_t unpack_nas_GetServingNetwork_t::MNC

8.554.2.6 uint8_t unpack_nas_GetServingNetwork_t::Name[255]

8.554.2.7 uint8_t unpack_nas_GetServingNetwork_t::nameSize

8.554.2.8 uint32_t unpack_nas_GetServingNetwork_t::PSDomain

8.554.2.9 uint8_t unpack_nas_GetServingNetwork_t::RadioIfaces[255]

8.554.2.10 uint8_t unpack_nas_GetServingNetwork_t::RadioIfacesSize

8.554.2.11 uint32_t unpack_nas_GetServingNetwork_t::RAN

8.554.2.12 uint32_t unpack_nas_GetServingNetwork_t::RegistrationState

8.554.2.13 uint32_t unpack_nas_GetServingNetwork_t::Roaming

8.555 unpack_nas_GetServingNetworkCapabilities_t Struct Reference

Data Fields

- uint8_t DataCapsLen
- uint8_t DataCaps [255]

8.555.1 Detailed Description

Parameters

DataCapsLen	• Upon input, the maximum number of elements the data capabilities array can contain. • Upon output, the actual number of elements in the data capabilities array.
DataCaps[OUT]	• Data capabilities array of unsigned long type – 1 - GPRS – 2 - EDGE – 3 - HSDPA – 4 - HSUPA – 5 - WCDMA – 6 - CDMA 1xRTT – 7 - CDMA 1xEV-DO Rev 0 – 8 - CDMA 1xEV-DO Rev. A – 9 - GSM – 10 - EVDO Rev. B – 11 - LTE – 12 - HSDPA Plus – 13 - Dual Carrier HSDPA Plus

8.555.2 Field Documentation

8.555.2.1 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCaps[255]

8.555.2.2 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCapsLen

8.556 unpack_nas_GetSignalStrengths_t Struct Reference

Data Fields

- uint32_t len
- signed char rssi [8]
- uint32_t radio [8]

8.556.1 Detailed Description

Parameters

len	number of rssi & radio items following
rssi	signal strength array
radio	radio interface array

8.556.2 Field Documentation

8.556.2.1 uint32_t unpack_nas_GetSignalStrengths_t::len

8.556.2.2 uint32_t unpack_nas_GetSignalStrengths_t::radio[8]

8.556.2.3 signed char unpack_nas_GetSignalStrengths_t::rssi[8]

8.557 unpack_nas_PerformNetworkScan_t Struct Reference

Data Fields

- uint8_t * p3GppNetworkInstanceSize
- nas_QmiNas3GppNetworkInfo * p3GppNetworkInfoInstances
- uint8_t * pRATInstanceSize
- nas_QmiNas3GppNetworkRAT * pRATINstance
- uint8_t * pPCSInstanceSize
- nas_QmisNasPcsDigit * pPCSInstance
- uint32_t * pScanResult

8.557.1 Detailed Description

Contain the network scan information.

Parameters

p3GppNetwork-InstanceSize	• Upon input, maximum number of elements that the network info instance array can contain. • Upon successful output, the actual number of elements in the network info instance array.
p3GppNetwork-InfoInstances	• Network info instance array – See nas_QmiNas3GppNetworkInfo for more information
pRATInstance-Size	• Upon input, maximum number of elements that the RAT info instance array can contain. • Upon successful output, the actual number of elements in the RAT info instance array.
pRATINstance	• RAT info instance array – See nas_QmiNas3GppNetworkRAT for more information
pPCSInstance-Size	• Upon input, maximum number of elements that the PCS Digit info instance array can contain. • Upon successful output, the actual number of elements in the PCS Digit info instance array.
pPCSInstance	• PCS Digit info instance array – See nas_QmisNasPcsDigit for more information
pScanResult	• status of network scan • 0x00 - scan successful • 0x01 - scan was aborted • 0x02 - scan did not complete due to a radio link failure recovery in progress

8.557.2 Field Documentation

8.557.2.1 nas_QmiNas3GppNetworkInfo* unpack_nas_PerformNetworkScan_t::p3GppNetworkInfoInstances

8.557.2.2 uint8_t* unpack_nas_PerformNetworkScan_t::p3GppNetworkInstanceSize

8.557.2.3 nas_QmisNasPcsDigit* unpack_nas_PerformNetworkScan_t::pPCSInstance

8.557.2.4 uint8_t* unpack_nas_PerformNetworkScan_t::pPCSInstanceSize

8.557.2.5 nas_QmiNas3GppNetworkRAT* unpack_nas_PerformNetworkScan_t::pRATINstance

8.557.2.6 uint8_t* unpack_nas_PerformNetworkScan_t::pRATInstanceSize

8.557.2.7 uint32_t* unpack_nas_PerformNetworkScan_t::pScanResult

8.558 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference

Data Fields

- uint8_t dataCapsSize
- uint8_t dataCaps [255]

8.558.1 Detailed Description

Parameters

dataCapsSize	Number of Data Capabilities
dataCaps	Data Capabilities

8.558.2 Field Documentation

8.558.2.1 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCaps[255]

8.558.2.2 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCapsSize

8.559 unpack_nas_SetEventReportInd_t Struct Reference

Data Fields

- nas_SignalStrengthTlv SSTlv
- nas_RFInfoTlv RFTlv
- nas_RejectReasonTlv RRTlv
- nas_SLQSSignalStrengthsTlv SLQSSSTlv

8.559.1 Detailed Description

Parameters

SSTlv	signal strength tlv
RFTlv	RF tlv
RRTlv	RR tlv
SLQSSSTlv	signal strength complete info tlv

8.559.2 Field Documentation

8.559.2.1 nas_RFInfoTlv unpack_nas_SetEventReportInd_t::RFTlv

8.559.2.2 nas_RejectReasonTlv unpack_nas_SetEventReportInd_t::RRTlv

8.559.2.3 nas_SLQSSignalStrengthsTlv unpack_nas_SetEventReportInd_t::SLQSSSTlv

8.559.2.4 nas_SignalStrengthTlv unpack_nas_SetEventReportInd_t::SSTlv

8.560 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference

Data Fields

- nas_PhyCaAggScellIndType sPhyCaAggScellIndType
- nas_PhyCaAggScellDlBw sPhyCaAggScellDlBw
- nas_PhyCaAggScellInfo sPhyCaAggScellInfo
- nas_PhyCaAggPcellInfo sPhyCaAggPcellInfo
- nas_PhyCaAggScellIndex sPhyCaAggScellIndex

8.560.1 Detailed Description

Structure for storing the LTEC PHY CA indication parameters.

Parameters

pPhyCaAgg-ScellIndType	• See nas_PhyCaAggScellIndType for more information.
sPhyCaAgg-ScellDlBw	• See nas_PhyCaAggScellDlBw for more information.
sPhyCaAgg-ScellInfo	• See nas_PhyCaAggScellInfo for more information.
sPhyCaAgg-PcellInfo	• See nas_PhyCaAggPcellInfo for more information.
sPhyCaAgg-ScellIndex	• See nas_PhyCaAggScellIndex for more information.

8.560.2 Field Documentation

8.560.2.1 nas_PhyCaAggPcellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggPcellInfo

8.560.2.2 nas_PhyCaAggScellDlBw unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellDlBw

8.560.2.3 nas_PhyCaAggScellIndex unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndex

8.560.2.4 nas_PhyCaAggScellIndType unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndType

8.560.2.5 nas_PhyCaAggScellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellInfo

8.561 unpack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint16_t TlvResult

8.561.1 Detailed Description

Parameters

TechnologyPref[-OUT]	• Bitmask representing the radio technology preference set. • No bits set indicates to the device to automatically determine the technology to use • Values: – Bit 0 - Technology is 3GPP2 – Bit 1 - Technology is 3GPP • Any combination of the following may be returned: – Bit 2 - Analog - AMPS if 3GPP2, GSM if 3GPP – Bit 3 - Digital - CDMA if 3GPP2, WCDMA if 3GPP – Bit 4 - HDR – Bit 5 - LTE – Bits 6 to 15 - Reserved
Duration[OUT]	• Duration of active preference – 0 - Permanent – 1 - Power cycle – 2 - Until the end of the next call or a power cycle – 3 - Until the end of the next call, a specified time, or a power cycle – 4 to 6 - Until the end of the next call
Persistent-TechnologyPref[-OUT]	• Bit field representing persistent radio technology preference – Same representation as the pTechnologyPref parameter
TlvResult	• unpack result

8.561.2 Field Documentation

8.561.2.1 uint16_t unpack_nas_SetNetworkPreference_t::TlvResult

8.562 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference

Data Fields

- uint8_t roaming

8.562.1 Detailed Description

Parameters

roaming	• Roaming Indication – 0 - Roaming – 1 - Home – 2 - Roaming partner – >2 - Operator defined values

8.562.2 Field Documentation

8.562.2.1 uint8_t unpack_nas_SetRoamingIndicatorCallback_ind_t::roaming

8.563 unpack_nas_SetServingSystemCallback_ind_t Struct Reference

Data Fields

- NASServingSystemInfo SSInfo
- uint16_t Tlvresult

8.563.1 Detailed Description

Parameters

SSInfo	• Serving system parameters information – See NASServingSystemInfo for more details
Tlvresult	• unpack result

8.563.2 Field Documentation

8.563.2.1 NASServingSystemInfo unpack_nas_SetServingSystemCallback_ind_t::SSInfo

8.563.2.2 uint16_t unpack_nas_SetServingSystemCallback_ind_t::Tlvresult

8.564 unpack_nas_SLQSGetErrorRate_t Struct Reference

Data Fields

- uint16_t * pCDMAFrameErrRate
- uint16_t * pHDRPackErrRate
- uint8_t * pGSMBER
- uint8_t * pWCDMABER

8.564.1 Detailed Description

Structure for Error Rate unpack parameters.

Parameters

pCDMAFrame-ErrRate	• CDMA Frame Error Rate • Valid error rate values between 1 and 10000 are returned to indicate the percentage, e.g., a value of 300 means the error rate is 3%. • A value of 0xFFFF indicates that the error rate is unknown/unavailable.
pHDRPackErr-Rate	• HDR Packet Error Rate • Valid error rate values between 1 and 10000 are returned to indicate the percentage, e.g., a value of 300 means the error rate is 3%. • A value of 0xFFFF indicates that the error rate is unknown/unavailable.
pGSMBER	• GSM Bit Error Rate • Valid error rate values between 1 and 100 are returned to indicate the percentage value. • A 0% block error rate (BLER) indicates No Data.
pWCDMABER	• WCDMA Block Error Rate • Valid error rate values between 1 and 100 are returned to indicate the percentage value. • A value of 0xFF indicates that the error rate is unknown/unavailable.

8.564.2 Field Documentation

8.564.2.1 uint16_t* unpack_nas_SLQSGetErrorRate_t::pCDMAFrameErrRate

8.564.2.2 uint8_t* unpack_nas_SLQSGetErrorRate_t::pGSMBER

8.564.2.3 uint16_t* unpack_nas_SLQSGetErrorRate_t::pHDRPackErrRate

8.564.2.4 uint8_t* unpack_nas_SLQSGetErrorRate_t::pWCDMABER

8.565 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference

Data Fields

- NasGetLTECphyCaInfo LTECphyCaInfo
- uint16_t Tlvresult

8.565.1 Detailed Description

Parameters

LTECphyCa	• Carrier aggregation event information – See NasGetLTECphyCaInfo for more details
Tlvresult	• unpack result

8.565.2 Field Documentation

8.565.2.1 NasGetLTECphyCaInfo unpack_nas_SlqsGetLTECphyCAInfo_t::LTECphyCaInfo

8.565.2.2 uint16_t unpack_nas_SlqsGetLTECphyCAInfo_t::TlvResult

8.566 unpack_nas_SLQSGetNetworkTime_t Struct Reference

Data Fields

- nas_timeInfo * p3GPP2TimeInfo
- nas_timeInfo * p3GPPTimeInfo

8.566.1 Detailed Description

This structure contains information about the GetNetworkTime response parameters.

Parameters

p3GPP2Time-Info	[Optional] • See nas_timeInfo for more information
p3GPPTimeInfo	[Optional] • See nas_timeInfo for more information

8.566.2 Field Documentation

8.566.2.1 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPP2TimeInfo

8.566.2.2 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPPTimeInfo

8.567 unpack_nas_SLQSGetOperatorNameData_t Struct Reference

Data Fields

- nas_serviceProviderName * pSrvcProviderName
- nas_operatorPLMNList * pOperatorPLMNList
- nas_PLMNNetworkName * pPLMNNetworkName
- nas_operatorNameString * pOperatorNameString
- nas_PLMNNetworkNameData * pNITZInformation

8.567.1 Detailed Description

Structure for Operator Name unpack.

Parameters

pSrvcProvider-Name	• Refer nas_serviceProviderName for details (Optional). • Can provide NULL if this parameter is not required.

pOperatorPLMNList	• Refer nas_operatorPLMNList for details (Optional). • Can provide NULL if this parameter is not required.
pPLMNNetworkName	• Refer nas_PLMNNetworkName for details (Optional). • Can provide NULL if this parameter is not required.
pOperatorNameString	• Refer nas_operatorNameString for details (Optional). • Can provide NULL if this parameter is not required.
pNITZInformation	• Refer nas_PLMNNetworkNameData for details (Optional). • Can provide NULL if this parameter is not required.

8.567.2 Field Documentation

8.567.2.1 nas_PLMNNetworkNameData∗ unpack_nas_SLQSGetOperatorNameData_t::pNITZInformation

8.567.2.2 nas_operatorNameString∗ unpack_nas_SLQSGetOperatorNameData_t::pOperatorNameString

8.567.2.3 nas_operatorPLMNList∗ unpack_nas_SLQSGetOperatorNameData_t::pOperatorPLMNList

8.567.2.4 nas_PLMNNetworkName∗ unpack_nas_SLQSGetOperatorNameData_t::pPLMNNetworkName

8.567.2.5 nas_serviceProviderName∗ unpack_nas_SLQSGetOperatorNameData_t::pSrvcProviderName

8.568 unpack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint8_t spnEncoding
- uint8_t spnLength
- char spn [255]
- uint8_t shortNameEn
- uint8_t shortNameCI
- uint8_t shortNameSB
- char shortNameLen
- uint8_t shortName [255]
- uint8_t longNameEn
- uint8_t longNameCI
- uint8_t longNameSB
- uint8_t longNameLen
- char longName [255]

8.568.1 Field Documentation

8.568.1.1 char unpack_nas_SLQSGetPLMNName_t::longName[255]

8.568.1.2 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameCI

8.568.1.3 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameEn

8.568.1.4 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameLen

8.568.1.5 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameSB

8.568.1.6 uint8_t unpack_nas_SLQSGetPLMNName_t::shortName[255]

8.568.1.7 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameCI

8.568.1.8 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameEn

8.568.1.9 char unpack_nas_SLQSGetPLMNName_t::shortNameLen

8.568.1.10 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameSB

8.568.1.11 char unpack_nas_SLQSGetPLMNName_t::spn[255]

8.568.1.12 uint8_t unpack_nas_SLQSGetPLMNName_t::spnEncoding

8.568.1.13 uint8_t unpack_nas_SLQSGetPLMNName_t::spnLength

8.569 unpack_nas_SLQSGetServingSystem_t Struct Reference

Data Fields

- nas_servSystem ServingSystem
- uint8_t RoamIndicatorVal
- nas_dataSrvCapabilities DataSrvCapabilities
- nas_currentPLMN CurrentPLMN
- uint16_t SystemID
- uint16_t NetworkID
- uint16_t BasestationID
- uint32_t BasestationLatitude
- uint32_t BasestationLongitude
- nas_roamIndList RoamingIndicatorList
- uint8_t DefaultRoamInd
- nas_qaQmi3Gpp2TimeZone Gpp2TimeZone
- uint8_t CDMA_P_Rev
- uint8_t GppTimeZone
- uint8_t GppNetworkDSTAdjustment
- uint16_t Lac
- uint32_t CellID
- uint8_t ConcSvcInfo
- uint8_t PRLInd
- uint8_t DTMInd
- nas_detailSvcInfo DetailedSvcInfo
- nas_CDMASysInfoExt CDMASystemInfoExt
- uint8_t HdrPersonality
- uint16_t TrackAreaCode
- nas_callBarStatus CallBarStatus

8.569.1 Detailed Description

This structure contains the Serving System parameters

- Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.

Parameters

ServingSystem	• Serving System • See nas_servSystem for more information
roamIndicatorVal	• Optional parameter indicating Roaming Indicator value • Values: – 0x00 - Roaming – 0x01 - Home – 0x02 - Flashing – 0x03 and above - Operator defined values
DataSrv-Capabilities	• Optional parameter indicating Data services capability • See nas_dataSrvCapabilities for more information
CurrentPLMN	• Optional parameter indicating Current PLMN • See nas_currentPLMN for more information
SystemID	• Optional parameter indicating System ID
NetworkID	• Optional parameter indicating Network ID
BaseStationID	• Optional parameter indicating Base Station Identification Number
BaseStation-Latitude	• Optional parameter indicating Base station latitude in units of 0.25 sec,expressed as a two's complement signed number with positive numbers signifying North latitude
Basestation-Longitude	• Optional parameter indicating Base station longitude in units of 0.25 sec, expressed as a Two's complement signed number with positive numbers signifying East longitude
Roaming-IndicatorList	• Optional parameter indicating Roaming Indicator List • See nas_roamIndList for more information
defaultRoamInd	• Optional parameter indicating Default Roaming Indicator • Values: – 0x00 - Roaming – 0x01 - Home

Gpp2TimeZone	• Optional parameter indicating 3GPP2 Time Zone • See nas_qaQmi3Gpp2TimeZone for more information
CDMA_P_Rev	• Optional parameter indicating CDMA P_Rev in use
GppTimeZone	• Optional parameter indicating Offset from Universal time, i.e., difference between local time and Universal time, in increments of 15 min. (signed value).
GppNetworkDS-TAdjustment	• Optional parameter indicating 3GPP network daylight saving adjustment • Values: – 0x00 - No adjustment for Daylight Saving Time – 0x01 - 1 hr adjustment for Daylight Saving Time – 0x02 - 2 hr adjustment for Daylight Saving Time
Lac	• Optional parameter indicating 3GPP Location Area Code
CellID	• Optional parameter indicating 3GPP Cell ID
concSvcInfo	• Optional parameter indicating 3GPP2 concurrent service Info • Values: – 0x00 - Concurrent service not available – 0x01 - Concurrent service available
PRLInd	• Optional parameter indicating 3GPP2 PRL Indicator • Values: – 0x00 - System not in PRL – 0x01 - System is in PRL
DTMInd	• Optional parameter indicating Dual Transfer Mode Indication(GSM Only) • Values: – 0x00 - DTM not supported – 0x01 - DTM supported
DetailedSvcInfo	• Optional parameter indicating Detailed service information • See nas_detailSvcInfo for more information
CDMASystem-InfoExt	• Optional parameter indicating CDMA System Info Ext • See nas_CDMASysInfoExt for more information

hdrPersonality	• Optional parameter indicating HDR Personality Information • Values: – 0x00 - Unknown – 0x01 - HRPD – 0x02 - eHRPD
trackAreaCode	• Optional parameter indicating Tracking area code information for LTE
CallBarStatus	• Optional parameter indicating Call Barring Status • See nas_callBarStatus for more information

8.569.2 Field Documentation

8.569.2.1 uint16_t unpack_nas_SLQSGetServingSystem_t::BasestationID

8.569.2.2 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLatitude

8.569.2.3 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLongitude

8.569.2.4 nas_callBarStatus unpack_nas_SLQSGetServingSystem_t::CallBarStatus

8.569.2.5 uint8_t unpack_nas_SLQSGetServingSystem_t::CDMA_P_Rev

8.569.2.6 nas_CDMASysInfoExt unpack_nas_SLQSGetServingSystem_t::CDMASystemInfoExt

8.569.2.7 uint32_t unpack_nas_SLQSGetServingSystem_t::CellID

8.569.2.8 uint8_t unpack_nas_SLQSGetServingSystem_t::ConcSvcInfo

8.569.2.9 nas_currentPLMN unpack_nas_SLQSGetServingSystem_t::CurrentPLMN

8.569.2.10 nas_dataSrvCapabilities unpack_nas_SLQSGetServingSystem_t::DataSrvCapabilities

8.569.2.11 uint8_t unpack_nas_SLQSGetServingSystem_t::DefaultRoamInd

8.569.2.12 nas_detailSvcInfo unpack_nas_SLQSGetServingSystem_t::DetailedSvcInfo

8.569.2.13 uint8_t unpack_nas_SLQSGetServingSystem_t::DTMInd

8.569.2.14 nas_qaQmi3Gpp2TimeZone unpack_nas_SLQSGetServingSystem_t::Gpp2TimeZone

8.569.2.15 uint8_t unpack_nas_SLQSGetServingSystem_t::GppNetworkDSTAdjustment

8.569.2.16 uint8_t unpack_nas_SLQSGetServingSystem_t::GppTimeZone

8.569.2.17 uint8_t unpack_nas_SLQSGetServingSystem_t::HdrPersonality

8.569.2.18 uint16_t unpack_nas_SLQSGetServingSystem_t::Lac

8.569.2.19 uint16_t unpack_nas_SLQSGetServingSystem_t::NetworkID

8.569.2.20 uint8_t unpack_nas_SLQSGetServingSystem_t::PRLInd

8.569.2.21 uint8_t unpack_nas_SLQSGetServingSystem_t::RoamIndicatorVal

8.569.2.22 nas_roamIndList unpack_nas_SLQSGetServingSystem_t::RoamingIndicatorList

8.569.2.23 nas_servSystem unpack_nas_SLQSGetServingSystem_t::ServingSystem

8.569.2.24 uint16_t unpack_nas_SLQSGetServingSystem_t::SystemID

8.569.2.25 uint16_t unpack_nas_SLQSGetServingSystem_t::TrackAreaCode

8.570 unpack_nas_SLQSGetSignalStrength_t Struct Reference

Data Fields

- uint16_t signalStrengthReqMask
- uint16_t rxSignalStrengthListLen
- nas_rxSignalStrengthListElement rxSignalStrengthList [18]
- uint16_t ecioListLen
- nas_ecioListElement ecioList [18]
- int32_t Io
- uint8_t sinr
- uint16_t errorRateListLen
- nas_errorRateListElement errorRateList [18]
- nas_rsrqInformation rsrqInfo
- int16_t ltesnr
- int16_t ltersrp

8.570.1 Detailed Description

This structure contains the Signal Strength Information

Parameters

signalStrength-ReqMask	• Request Mask – Request additional signal information for: Bit 0 - RSSI Information bit Valid values are: 0 - Do Not Request Additional Info for RSSI 1 - Request Additional Info for RSSI Bit 1 - ECIO Information bit Valid values are: 0 - Do Not Request Additional Info for ECIO 1 - Request Additional Info for ECIO Bit 2 - IO Information bit Valid values are: 0 - Do Not Request Additional Info for IO 1 - Request Additional Info for IO Bit 3 - SINR Information bit Valid values are: 0 - Do Not Request Additional Info for SINR 1 - Request Additional Info for SINR Bit 4 - ERROR RATE Information bit Valid values are: 0 - Do Not Request Additional Info for Error Rate 1 - Request Additional Info for Error Rate Bit 5 - RSRQ Information bit Valid values are: 0 - Do Not Request Additional Info for RSRQ 1 - Request Additional Info for RSRQ Bit 6 - LTE SNR information bit Valid values are: 0 - Do not request additional information for LTE SNR 1 - Request additional information for LTE SNR Bit 7 - LTE RSRP Information bit Valid values are: 0 - Do not request additional information for LTE RSRP 1 - Request additional information for LTE RSRP
rxSignal-StrengthListLen	• Number of elements in Receive Signal Strength List
rxSignal-StrengthList	• See nas_rxSignalStrengthListElement for more information
ecioListLen	• Number of elements in ECIO List
ecioList	• See nas_ecioListElement for more information
Io	• Received Io in dBm; IO is only applicable for 1xEV-DO
sinr	• SINR level – SINR is only applicable for 1xEV-DO; valid levels are 0 to 8 where maximum value for 0 - SINR_LEVEL_0 is -9 dB 1 - SINR_LEVEL_1 is -6 dB 2 - SINR_LEVEL_2 is -4.5 dB 3 - SINR_LEVEL_3 is -3 dB 4 - SINR_LEVEL_4 is -2 dB 5 - SINR_LEVEL_5 is +1 dB 6 - SINR_LEVEL_6 is +3 dB 7 - SINR_LEVEL_7 is +6 dB 8 - SINR_LEVEL_8 is +9 dB

errorRateListLen	• Number of elements in Error Rate List
errorRateList	• See nas_errorRateListElement for more information
rsrqInfo	• See nas_rsrqInformation for more information
ltesnr	• LTE SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246. LTE SNR is included only when the current serving system is LTE
ltersrp	• LTE SNR level as a scaled integer in units of 0.1 dB; e.g., -16 dB has a value of -160 and 24.6 dB has a value of 246. LTE SNR is included only when the current serving system is LTE

8.570.2 Field Documentation

8.570.2.1 nas_ecioListElement unpack_nas_SLQSGetSignalStrength_t::ecioList[18]

8.570.2.2 uint16_t unpack_nas_SLQSGetSignalStrength_t::ecioListLen

8.570.2.3 nas_errorRateListElement unpack_nas_SLQSGetSignalStrength_t::errorRateList[18]

8.570.2.4 uint16_t unpack_nas_SLQSGetSignalStrength_t::errorRateListLen

8.570.2.5 int32_t unpack_nas_SLQSGetSignalStrength_t::lo

8.570.2.6 int16_t unpack_nas_SLQSGetSignalStrength_t::ltersrp

8.570.2.7 int16_t unpack_nas_SLQSGetSignalStrength_t::ltesnr

8.570.2.8 nas_rsrqInformation unpack_nas_SLQSGetSignalStrength_t::rsrqInfo

8.570.2.9 nas_rxSignalStrengthListElement unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthList[18]

8.570.2.10 uint16_t unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthListLen

8.570.2.11 uint16_t unpack_nas_SLQSGetSignalStrength_t::signalStrengthReqMask

8.570.2.12 uint8_t unpack_nas_SLQSGetSignalStrength_t::sinr

8.571 unpack_nas_SLQSGetSysInfo_t Struct Reference

Data Fields

- nas_SrvStatusInfo * pCDMASrvStatusInfo
- nas_SrvStatusInfo * pHDRSrvStatusInfo
- nas_GSMSrvStatusInfo * pGSMSrvStatusInfo
- nas_GSMSrvStatusInfo * pWCDMASrvStatusInfo
- nas_GSMSrvStatusInfo * pLTESrvStatusInfo

- nas_CDMASysInfo * pCDMASysInfo
- nas_HDRSysInfo * pHDRSysInfo
- nas_GSMSysInfo * pGSMSysInfo
- nas_WCDMASysInfo * pWCDMASysInfo
- nas_LTESysInfo * pLTESysInfo
- nas_AddCDMASysInfo * pAddCDMASysInfo
- uint16_t * pAddHDRSysInfo
- nas_AddSysInfo * pAddGSMSysInfo
- nas_AddSysInfo * pAddWCDMASysInfo
- uint16_t * pAddLTESysInfo
- nas_CallBarringSysInfo * pGSMCallBarringSysInfo
- nas_CallBarringSysInfo * pWCDMACallBarringSysInfo
- uint8_t * pLTEVoiceSupportSysInfo
- uint8_t * pGSMCipherDomainSysInfo
- uint8_t * pWCDMACipherDomainSysInfo

8.571.1 Detailed Description

Parameters

pCDMASrvStatusInfo	• See nas_SrvStatusInfo for more information.
pHDRSrvStatusInfo	• See nas_SrvStatusInfo for more information.
pGSMSrvStatusInfo	• See nas_GSMSrvStatusInfo for more information.
pWCDMASrvStatusInfo	• See nas_GSMSrvStatusInfo for more information.
pLTESrvStatusInfo	• See nas_GSMSrvStatusInfo for more information.
pCDMASysInfo	• See nas_CDMASysInfo for more information.
pHDRSysInfo	• See nas_HDRSysInfo for more information.
pGSMSysInfo	• See nas_GSMSysInfo for more information.
pWCDMASysInfo	• See nas_WCDMASysInfo for more information.
pLTESysInfo	• See nas_LTESysInfo for more information.
pAddCDMASysInfo	• See nas_AddCDMASysInfo for more information.

pAddHDRSysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
pAddGSMSysInfo	• See nas_AddSysInfo for more information.
pAddWCDMASysInfo	• See nas_AddSysInfo for more information.
pAddLTESysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
pGSMCallBarringSysInfo	• See nas_CallBarringSysInfo for more information.
pWCDMACallBarringSysInfo	• See nas_CallBarringSysInfo for more information.
pLTEVoiceSupportSysInfo	• Indicates voice support status on LTE. – 0x00 - Voice is not supported – 0x01 - Voice is supported
pGSMCipherDomainSysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
pWCDMACipherDomainSysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched

8.571.2 Field Documentation

8.571.2.1 nas_AddCDMASysInfo* unpack_nas_SLQSGetSysInfo_t::pAddCDMASysInfo

8.571.2.2 nas_AddSysInfo* unpack_nas_SLQSGetSysInfo_t::pAddGSMSysInfo

8.571.2.3 uint16_t* unpack_nas_SLQSGetSysInfo_t::pAddHDRSysInfo

8.571.2.4 uint16_t* unpack_nas_SLQSGetSysInfo_t::pAddLTESysInfo

8.571.2.5 **nas_AddSysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pAddWCDMASysInfo

8.571.2.6 **nas_SrvStatusInfo**∗ unpack_nas_SLQSGetSysInfo_t::pCDMASrvStatusInfo

8.571.2.7 **nas_CDMASysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pCDMASysInfo

8.571.2.8 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pGSMCallBarringSysInfo

8.571.2.9 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pGSMCipherDomainSysInfo

8.571.2.10 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSGetSysInfo_t::pGSMSrvStatusInfo

8.571.2.11 **nas_GSMSysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pGSMSysInfo

8.571.2.12 **nas_SrvStatusInfo**∗ unpack_nas_SLQSGetSysInfo_t::pHDRSrvStatusInfo

8.571.2.13 **nas_HDRSysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pHDRSysInfo

8.571.2.14 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSGetSysInfo_t::pLTESrvStatusInfo

8.571.2.15 **nas_LTESysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pLTESysInfo

8.571.2.16 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pLTEVoiceSupportSysInfo

8.571.2.17 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pWCDMACallBarringSysInfo

8.571.2.18 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pWCDMACipherDomainSysInfo

8.571.2.19 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSGetSysInfo_t::pWCDMASrvStatusInfo

8.571.2.20 **nas_WCDMASysInfo**∗ unpack_nas_SLQSGetSysInfo_t::pWCDMASysInfo

8.572 unpack_nas_SLQSGetSysSelectionPref_t Struct Reference

Data Fields

- uint8_t ∗ pEmerMode
- uint16_t ∗ pModePref
- uint64_t ∗ pBandPref
- uint16_t ∗ pPRLPref
- uint16_t ∗ pRoamPref
- uint64_t ∗ pLTEBandPref
- uint8_t ∗ pNetSelPref
- uint32_t ∗ pSrvDomainPref
- uint32_t ∗ pGWAcqOrderPref

8.572.1 Detailed Description

Parameters

pEmerMode	• Optional parameter specifying the emergency Mode • Values: – 0 - OFF (normal) – 1 - ON (Emergency)

pModePref	• Optional parameter • Bit Mask indicating the radio technology mode preference • Bit values: – Bit 0 - cdma2000 1x – Bit 1 - cdma2000 HRPD(1xEV-DO) – Bit 2 - GSM – Bit 3 - UMTS – Bit 4 - LTE

pBandPref	• Optional parameter • Bit mask representing the band preference • Bit values: – Bit 0 - Band Class 0, A-System – Bit 1 - Band Class 0, B-System, Band Class 0 AB, GSM 850 Band – Bit 2 - Band Class 1, all blocks – Bit 3 - Band Class 2 place holder – Bit 4 - Band Class 3, A-System – Bit 5 - Band Class 4, all blocks – Bit 6 - Band Class 5, all blocks – Bit 7 - GSM_DCS_1800 band – Bit 8 - GSM Extended GSM (E-GSM) 900 band – Bit 9 - GSM Primary GSM (P-GSM) 900 band – Bit 10 - Band Class 6 – Bit 11 - Band Class 7 – Bit 12 - Band Class 8 – Bit 13 - Band Class 9 – Bit 14 - Band Class 10 – Bit 15 - Band Class 11 – Bit 16 - GSM 450 band – Bit 17 - GSM 480 band – Bit 18 - GSM 750 band – Bit 19 - GSM 850 band – Bit 20 - GSM Railways GSM 900 Band – Bit 21 - GSM PCS 1900 band – Bit 22 - WCDMA Europe, Japan, and China IMT 2100 band – Bit 23 - WCDMA U.S. PCS 1900 band – Bit 24 - WCDMA Europe and China DCS 1800 band – Bit 25 - WCDMA U.S. 1700 band – Bit 26 - WCDMA U.S. 850 band – Bit 27 - WCDMA Japan 800 band – Bit 28 - Band Class 12 – Bit 29 - Band Class 14 – Bit 30 - Reserved – Bit 31 - Band Class 15 – Bit 32 to 47 - Reserved – Bit 48 - WCDMA Europe 2600 band – Bit 49 - WCDMA Europe and Japan 900 band – Bit 50 - WCDMA Japan 1700 band – Bit 51 to 55 - Reserved – Bit 56 - Band Class 16 – Bit 57 - Band Class 17 – Bit 58 - Band Class 18 – Bit 59 - Band Class 19 – Bit 60 to 64 - Reserved

pPRLPref	• Optional parameter indicating the CDMA PRL Preference • Values: – 0x0001 - Acquire available system only on the A side – 0x0002 - Acquire available system only on the B side – 0x3FFF - Acquire any available systems
pRoamPref	• Optional parameter indicating the roaming Preference • Values: – 0x01 - Acquire only systems for which the roaming indicator is off – 0x02 - Acquire a system as long as its roaming indicator is not off – 0x03 - Acquire only systems for which the roaming indicator is off or solid on, i.e. not flashing; CDMA only – 0xFF - Acquire systems, regardless of their roaming indicator
pLTEBandPref	• Optional parameter • Bit mask representing the LTE band preference • Bit Values – Bit 0 - E-UTRA Operating Band 1 – Bit 1 - E-UTRA Operating Band 2 – Bit 2 - E-UTRA Operating Band 3 – Bit 3 - E-UTRA Operating Band 4 – Bit 4 - E-UTRA Operating Band 5 – Bit 5 - E-UTRA Operating Band 6 – Bit 6 - E-UTRA Operating Band 7 – Bit 7 - E-UTRA Operating Band 8 – Bit 8 - E-UTRA Operating Band 9 – Bit 9 - E-UTRA Operating Band 10 – Bit 10 - E-UTRA Operating Band 11 – Bit 11 - E-UTRA Operating Band 12 – Bit 12 - E-UTRA Operating Band 13 – Bit 13 - E-UTRA Operating Band 14 – Bit 16 - E-UTRA Operating Band 17 – Bit 17 - E-UTRA Operating Band 18 – Bit 18 - E-UTRA Operating Band 19 – Bit 19 - E-UTRA Operating Band 20 – Bit 20 - E-UTRA Operating Band 21 – Bit 32 - E-UTRA Operating Band 33 – Bit 33 - E-UTRA Operating Band 34 – Bit 34 - E-UTRA Operating Band 35 – Bit 35 - E-UTRA Operating Band 36 – Bit 36 - E-UTRA Operating Band 37 – Bit 37 - E-UTRA Operating Band 38 – Bit 38 - E-UTRA Operating Band 39 – Bit 39 - E-UTRA Operating Band 40 – All other bits are reserved

pNetSelPref	• Optional parameter indicating network selection preference • Values: – 0x00 - Automatic network selection – 0x01 - Manual network selection.
pSrvDomainPref	• Optional parameter indicating Service domain preference • Values: – 0x00 - Circuit switched only – 0x01 - Packet switched only – 0x02 - Circuit switched and packet switched – 0x03 - Packet switched attach – 0x04 - Packet switched detach
pGWAcqOrder-Pref	• Optional parameter indicating GSM/WCDMA Acquisition order Preference • Values: – 0x00 - Automatic – 0x01 - GSM then WCDMA – 0x02 - WCDMA then GSM

8.572.2 Field Documentation

8.572.2.1 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pBandPref

8.572.2.2 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pEmerMode

8.572.2.3 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pGWAcqOrderPref

8.572.2.4 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pLTEBandPref

8.572.2.5 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pModePref

8.572.2.6 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pNetSelPref

8.572.2.7 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pPRLPref

8.572.2.8 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pRoamPref

8.572.2.9 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pSrvDomainPref

8.573 unpack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference

Data Fields

- nas_namName * pNAMNameInfo
- nas_dirNum * pDirNum
- nas_homeSIDNID * pHomeSIDNID
- nas_minBasedIMSI * pMinBasedIMSI
- nas_trueIMSI * pTrueIMSI

- nas_CDMAChannel * pCDMAChannel

8.573.1 Detailed Description

Structure for 3GPP2Subscription unpack.

Parameters

pNAMNameInfo	[Optional] • See nas_namName for more information
pDirNum	[Optional] • See nas_dirNum for more information
pHomeSIDNID	[Optional] • See nas_homeSIDNID for more information
pMinBasedIMSI	[Optional] • See nas_minBasedIMSI for more information
pTrueIMSI	[Optional] • See nas_trueIMSI for more information
pCDMAChannel	[Optional] • See nas_CDMAChannel for more information

8.573.2 Field Documentation

8.573.2.1 nas_CDMAChannel* unpack_nas_SLQSNasGet3GPP2Subscription_t::pCDMAChannel

8.573.2.2 nas_dirNum* unpack_nas_SLQSNasGet3GPP2Subscription_t::pDirNum

8.573.2.3 nas_homeSIDNID* unpack_nas_SLQSNasGet3GPP2Subscription_t::pHomeSIDNID

8.573.2.4 nas_minBasedIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pMinBasedIMSI

8.573.2.5 nas_namName* unpack_nas_SLQSNasGet3GPP2Subscription_t::pNAMNameInfo

8.573.2.6 nas_trueIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pTrueIMSI

8.574 unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference

Data Fields

- nas_GERANInfo * pGERANInfo
- nas_UMTSInfo * pUMTSInfo
- nas_CDMAInfo * pCDMAInfo
- nas_LTEInfoIntrafreq * pLTEInfoIntrafreq
- nas_LTEInfoInterfreq * pLTEInfoInterfreq
- nas_LTEInfoNeighboringGSM * pLTEInfoNeighboringGSM
- nas_LTEInfoNeighboringWCDMA * pLTEInfoNeighboringWCDMA
- uint32_t * pUMTSCellID
- nas_WCDMAInfoLTENeighborCell * pWCDMAInfoLTENeighborCell

8.574.1 Detailed Description

This structure contains information about the Get Cell Location response parameters.

Parameters

pGERANInfo	• See nas_GERANInfo for more information.
pUMTSInfo	• See nas_UMTSInfo for more information.
pCDMAInfo	• See nas_CDMAInfo for more information.
pLTEInfo-Intrafreq	• See nas_LTEInfoIntrafreq for more information.
pLTEInfo-Interfreq	• See nas_LTEInfoInterfreq for more information.
pLTEInfo-NeighboringGSM	• See nas_LTEInfoNeighboringGSM for more information.
pLTEInfo-NeighboringWCDMA	• See nas_LTEInfoNeighboringWCDMA for more information.
pUMTSCellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
pWCDMAInfoLTENeighborCell	• See nas_WCDMAInfoLTENeighborCell for more information.

8.574.2 Field Documentation

8.574.2.1 nas_CDMAInfo∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pCDMAInfo

8.574.2.2 nas_GERANInfo∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pGERANInfo

8.574.2.3 nas_LTEInfoInterfreq∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoInterfreq

8.574.2.4 nas_LTEInfoIntrafreq∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoIntrafreq

8.574.2.5 nas_LTEInfoNeighboringGSM∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringGSM

8.574.2.6 nas_LTEInfoNeighboringWCDMA∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringWCDMA

8.574.2.7 uint32_t∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSCellID

8.574.2.8 nas_UMTSInfo∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSInfo

8.574.2.9 nas_WCDMAInfoLTENeighborCell∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pWCDMAInfoLTENeighborCell

8.575 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference

Data Fields

- uint8_t * pColorCode

8.575.1 Detailed Description

Structure for storing the current preferred system selection settings for the device.

Parameters

pColorCode	[Optional] • Color code value • Color code corresponding to the sector to which the AT is sending the access probe • See 3GPP2 C.S0024-B V3.0, Section 7.11.6.2.1 for more information. – 0xFF - Not Available

8.575.2 Field Documentation

8.575.2.1 uint8_t* unpack_nas_SLQSNasGetHDRColorCode_t::pColorCode

8.576 unpack_nas_SLQSNasGetSigInfo_t Struct Reference

Data Fields

- cdmaSSInfo CDMASSInfo
- hdrSSInfo HDRSSInfo
- int8_t GSMSSInfo
- cdmaSSInfo WCDMASSInfo
- lteSSInfo LTESSInfo

8.576.1 Detailed Description

This structure contains the SLQSNasGetSigInfo response parameters.

Parameters

pCDMASSInfo	[Optional] • See cdmaSSInfo for more information
pHDRSSInfo	[Optional] • See hdrSSInfo for more information
pGSMSSInfo	[Optional] • GSM signal strength is the RSSI in dBm (signed value). • A value of -125 dBm or lower is used to indicate No Signal.
pWCDMASSInfo	[Optional] • See cdmaSSInfo for more information

pLTESSInfo	[Optional] • See lteSSInfo for more information

8.576.2 Field Documentation

8.576.2.1 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::CDMASSInfo

8.576.2.2 int8_t unpack_nas_SLQSNasGetSigInfo_t::GSMSSInfo

8.576.2.3 hdrSSInfo unpack_nas_SLQSNasGetSigInfo_t::HDRSSInfo

8.576.2.4 lteSSInfo unpack_nas_SLQSNasGetSigInfo_t::LTESSInfo

8.576.2.5 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::WCDMASSInfo

8.577 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- nas_rxInfo * pRXChain0Info
- nas_rxInfo * pRXChain1Info
- nas_txInfo * pTXInfo

8.577.1 Detailed Description

Structure for TxRx Info unpack

Parameters

pRXChain0Info	[Optional] • See nas_rxInfo for more information.
pRXChain1Info	[Optional] • See nas_rxInfo for more information.
pTXInfo	[Optional] • See nas_txInfo for more information.

8.577.2 Field Documentation

8.577.2.1 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain0Info

8.577.2.2 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain1Info

8.577.2.3 nas_txInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pTXInfo

8.578 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference

Data Fields

- nas_UniversalTime universalTime
- uint8_t * pTimeZone
- uint8_t * pDayltSavAdj
- uint8_t * pRadioInterface

8.578.1 Detailed Description

Structure for storing the NAS Network Time indication parameters.

Parameters

universalTime	• See nas_UniversalTime for more information.
pTimeZone	• Time Zone. • Offset from Universal time, i.e., the difference between local time and Universal time, in increments of 15 min (signed value).
pDayltSavAdj	• Daylight Saving Adjustment. • Daylight saving adjustment in hr. – Possible values: 0, 1, and 2.
pRadioInterface	• Radio interface from which the information comes • Values – 0x01 - NAS_RADIO_IF_CDMA_1X - cdma2000 1X – 0x02 - NAS_RADIO_IF_CDMA_1XEVDO - cdma2000 HRPD (1xEV-DO) – 0x04 - NAS_RADIO_IF_GSM - GSM – 0x05 - NAS_RADIO_IF_UMTS - UMTS – 0x08 - NAS_RADIO_IF_LTE - LTE – 0x09 - NAS_RADIO_IF_TDSCDMA -TD-SCDMA

8.578.2 Field Documentation

8.578.2.1 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pDayltSavAdj

8.578.2.2 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pRadioInterface

8.578.2.3 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pTimeZone

8.578.2.4 nas_UniversalTime unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::universalTime

8.579 unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference

Data Fields

- cdmaSSInfo * pCDMASigInfo
- hdrSSInfo * pHDRSigInfo

- int8_t * pGSMSigInfo
- cdmaSSInfo * pWCDMASigInfo
- lteSSInfo * pLTESigInfo
- int8_t * pRscp
- tdscdmaSigInfoExt * pTDSCDMASigInfoExt

8.579.1 Detailed Description

Structure for storing the nasSigInfo indication parameters.

Parameters

pCDMASigInfo	• See cdmaSSInfo for more information.
pHDRSigInfo	• See hdrSSInfo for more information.
pGSMSigInfo	• one byte value, GSM signal strength is the RSSI in dBm (signed value). A value of -125 dBm or lower is used to indicate No Signal
pWCDMASigInfo	• See cdmaSSInfo for more information.
pLTESigInfo	• See lteSSInfo for more information.
pRscp	• RSCP of the Primary Common Control Physical Channel (PCCPCH) in dBm. Measurement range: -120 dBm to -25 dBm.
pTDSCDMASig-InfoExt	• See tdscdmaSigInfoExt for more information.

8.579.2 Field Documentation

8.579.2.1 cdmaSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pCDMASigInfo

8.579.2.2 int8_t* unpack_nas_SLQSNasSigInfoCallback_ind_t::pGSMSigInfo

8.579.2.3 hdrSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pHDRSigInfo

8.579.2.4 lteSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pLTESigInfo

8.579.2.5 int8_t* unpack_nas_SLQSNasSigInfoCallback_ind_t::pRscp

8.579.2.6 tdscdmaSigInfoExt* unpack_nas_SLQSNasSigInfoCallback_ind_t::pTDSCDMASigInfoExt

8.579.2.7 cdmaSSInfo* unpack_nas_SLQSNasSigInfoCallback_ind_t::pWCDMASigInfo

8.580 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN * pWcdmaUARFCN
- nas_lteEARFCN * pLteEARFCN
- nas_ltePCI * pLtePCI

8.580.1 Detailed Description

Structure for Get Channel Lock unpack.

Parameters

pWcdmaUARFCN	[Optional] • See nas_wcdmaUARFCN for more information
pLteEARFCN	[Optional] • See nas_lteEARFCN for more information
pLtePCI	[Optional] • See nas_ltePCI for more information

8.580.2 Field Documentation

8.580.2.1 nas_lteEARFCN* unpack_nas_SLQSNASSwiGetChannelLock_t::pLteEARFCN

8.580.2.2 nas_ltePCI* unpack_nas_SLQSNASSwiGetChannelLock_t::pLtePCI

8.580.2.3 nas_wcdmaUARFCN* unpack_nas_SLQSNASSwiGetChannelLock_t::pWcdmaUARFCN

8.581 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference

Data Fields

- nas_CommInfo commonInfo
- nas_LTEInfo * pLTEInfo

8.581.1 Detailed Description

Structure for storing the SLQS Nas Swi Modem Status response parameters.

Parameters

commonInfo	(mandatory) • See nas_CommInfo for more information
pLTEInfo	(optional) • See nas_LTEInfo for more information

8.581.2 Field Documentation

8.581.2.1 nas_CommInfo unpack_nas_SLQSNasSwiModemStatus_t::commonInfo

8.581.2.2 nas_LTEInfo* unpack_nas_SLQSNasSwiModemStatus_t::pLTEInfo

8.582 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSwiOTAMessageInd Info
- uint16_t Tlvresult

8.582.1 Detailed Description

Parameters

Info	• Structure used to store all QMI Notification Info. – See NASQmiCbkNasSwiOTAMessageInd for more details
Tlvresult	• unpack result

8.582.2 Field Documentation

8.582.2.1 NASQmiCbkNasSwiOTAMessageInd unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Info

8.582.2.2 uint16_t unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Tlvresult

8.583 unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference

Data Fields

- char t3396_apn [101]
- uint8_t t3396_plmn_id [3]
- uint32_t t3396_val

8.583.1 Detailed Description

Structure for Network Timer indication parameters.

Parameters

t3396_apn	• apn
t3396_plmn_id	• plmn id
t3396_val	• timer value

8.583.2 Field Documentation

8.583.2.1 char unpack_nas_SLQSNasTimerCallback_ind_t::t3396_apn[101]

8.583.2.2 uint8_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_plmn_id[3]

8.583.2.3 uint32_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_val

8.584 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSystemSelPrefInd Info
- uint16_t Tlvresult

8.584.1 Detailed Description

Parameters

Info	• Structure used to store all QMI Notification Info. – See NASQmiCbkNasSystemSelPrefInd for more details
Tlvresult	• unpack result

8.584.2 Field Documentation

8.584.2.1 NASQmiCbkNasSystemSelPrefInd unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Info

8.584.2.2 uint16_t unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Tlvresult

8.585 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference

Data Fields

- uint16_t * pCurrentPersonality
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtocolSubtypeElement

8.585.1 Detailed Description

Structure for HDRPersonality unpack.

Parameters

pCurrent-Personality[Out]	• Current active personality index.
pPersonalityList-Length[In/Out]	• Number of Personality Protocol Subtype contains in this response. • maximum input value is 3

pProtocol-Subtype-Element[Out]	• See nas_protocolSubtypeElement for more information.

8.585.2 Field Documentation

8.585.2.1 uint16_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pCurrentPersonality

8.585.2.2 uint8_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pPersonalityListLength

8.585.2.3 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRPersonality_t::pProtocolSubtypeElement

8.586 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference

Data Fields

- uint16_t * pCurrentPrsnlty
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtoSubTypElmnt
- uint64_t * pAppSubType

8.586.1 Detailed Description

Structure for HDRProtSubtype unpack.

Parameters

pCurrent-Personality	• Current active personality index.
pPersonalityList-Length	• Number of Personality Protocol Subtype contains in this response. • maximum input value is 4
pProtocol-SubtypeElement	• See nas_protocolSubtypeElement for more information.
pAppSubType	• Stream application subtype • Application subtype for each stream,

8.586.2 Field Documentation

8.586.2.1 uint64_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pAppSubType

8.586.2.2 uint16_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pCurrentPrsnlty

8.586.2.3 uint8_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pPersonalityListLength

8.586.2.4 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pProtoSubTypElmnt

8.587 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference

Data Fields

- nas_DRCParams * pDRCParams
- uint8_t * pUATI
- nas_PilotSetData * pPilotSetData

8.587.1 Detailed Description

Structure for HRPD Stats unpack.

Parameters

pDRCParams	• See nas_DRCParams for more information.
pUATI	• A 128-bit address that includes the access terminal identifier and subnet ID • Size must be 16 bytes
pPilotSetData	• See nas_PilotSetData for more information.

8.587.2 Field Documentation

8.587.2.1 nas_DRCParams* unpack_nas_SLQSSwiGetHRPDStats_t::pDRCParams

8.587.2.2 nas_PilotSetData* unpack_nas_SLQSSwiGetHRPDStats_t::pPilotSetData

8.587.2.3 uint8_t* unpack_nas_SLQSSwiGetHRPDStats_t::pUATI

8.588 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference

Data Fields

- uint8_t ValidityCW0
- uint8_t CQIValueCW0
- uint8_t ValidityCW1
- uint8_t CQIValueCW1

8.588.1 Detailed Description

Parameters

ValidityCW0[OUT]	• Values – 0- Invalid. – 1- Valid.

CQIValueCW0[-OUT]	• Values – Range 0∼15
ValidityCW1[OUT]	• Values – 0- Invalid. – 1- Valid.
CQIValueCW1[-OUT]	• Values – Range 0∼15

8.588.2 Field Documentation

8.588.2.1 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW0

8.588.2.2 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW1

8.588.2.3 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW0

8.588.2.4 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW1

8.589 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference

Data Fields

- nas_SccRxInfo ∗ pSccRxInfo

8.589.1 Detailed Description

Parameters

pSccRxInfo	Secondary carrier Rx signal level info

8.589.2 Field Documentation

8.589.2.1 nas_SccRxInfo∗ unpack_nas_SLQSSwiGetLteSccRxInfo_t::pSccRxInfo

8.590 unpack_nas_SLQSSwiNetworkDebug_t Struct Reference

Data Fields

- uint8_t ∗ pObjectVer
- nas_NetworkStat1x ∗ pNetworkStat1x
- nas_NetworkStatEVDO ∗ pNetworkStatEVDO
- nas_DeviceConfigDetail ∗ pDeviceConfigDetail
- nas_DataStatusDetail ∗ pDataStatusDetail

8.590.1 Detailed Description

Structure for SwiNetworkDebug unpack.

Parameters

pObjectVer	• Object's version number for the host to handle – 0xFF - NA – Others - shows in decimal
pNetworkStat1x	• See nas_NetworkStat1x for more information
pNetworkStatE-VDO	• See nas_NetworkStatEVDO for more information.
pDeviceConfig-Detail	• See nas_DeviceConfigDetail for more information.
pDataStatus-Detail	• See nas_DataStatusDetail for more information.

8.590.2 Field Documentation

8.590.2.1 nas_DataStatusDetail∗ unpack_nas_SLQSSwiNetworkDebug_t::pDataStatusDetail

8.590.2.2 nas_DeviceConfigDetail∗ unpack_nas_SLQSSwiNetworkDebug_t::pDeviceConfigDetail

8.590.2.3 nas_NetworkStat1x∗ unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStat1x

8.590.2.4 nas_NetworkStatEVDO∗ unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStatEVDO

8.590.2.5 uint8_t∗ unpack_nas_SLQSSwiNetworkDebug_t::pObjectVer

8.591 unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference

Data Fields

- nas_SrvStatusInfo ∗ pCDMASrvStatusInfo
- nas_SrvStatusInfo ∗ pHDRSrvStatusInfo
- nas_GSMSrvStatusInfo ∗ pGSMSrvStatusInfo
- nas_GSMSrvStatusInfo ∗ pWCDMASrvStatusInfo
- nas_GSMSrvStatusInfo ∗ pLTESrvStatusInfo
- nas_CDMASysInfo ∗ pCDMASysInfo
- nas_HDRSysInfo ∗ pHDRSysInfo
- nas_GSMSysInfo ∗ pGSMSysInfo
- nas_WCDMASysInfo ∗ pWCDMASysInfo
- nas_LTESysInfo ∗ pLTESysInfo
- nas_AddCDMASysInfo ∗ pAddCDMASysInfo
- uint16_t ∗ pAddHDRSysInfo
- nas_AddSysInfo ∗ pAddGSMSysInfo
- nas_AddSysInfo ∗ pAddWCDMASysInfo

- uint16_t * pAddLTESysInfo
- nas_CallBarringSysInfo * pGSMCallBarringSysInfo
- nas_CallBarringSysInfo * pWCDMACallBarringSysInfo
- uint8_t * pLTEVoiceSupportSysInfo
- uint8_t * pGSMCipherDomainSysInfo
- uint8_t * pWCDMACipherDomainSysInfo
- uint8_t * pSysInfoNoChange

8.591.1 Detailed Description

Parameters

pCDMASrvStatusInfo	• See nas_SrvStatusInfo for more information.
pHDRSrvStatusInfo	• See nas_SrvStatusInfo for more information.
pGSMSrvStatusInfo	• See nas_GSMSrvStatusInfo for more information.
pWCDMASrvStatusInfo	• See nas_GSMSrvStatusInfo for more information.
pLTESrvStatusInfo	• See nas_GSMSrvStatusInfo for more information.
pCDMASysInfo	• See nas_CDMASysInfo for more information.
pHDRSysInfo	• See nas_HDRSysInfo for more information.
pGSMSysInfo	• See nas_GSMSysInfo for more information.
pWCDMASysInfo	• See nas_WCDMASysInfo for more information.
pLTESysInfo	• See nas_LTESysInfo for more information.
pAddCDMASysInfo	• See nas_AddCDMASysInfo for more information.
pAddHDRSysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
pAddGSMSysInfo	• See nas_AddSysInfo for more information.

pAddWCDMA-SysInfo	• See nas_AddSysInfo for more information.
pAddLTESysInfo	• System table index referencing the beginning of the geo in which the current serving system is present. • When the system index is not known, 0xFFFF is used.
pGSMCall-BarringSysInfo	• See nas_CallBarringSysInfo for more information.
pWCDMACall-BarringSysInfo	• See nas_CallBarringSysInfo for more information.
pLTEVoice-SupportSysInfo	• Indicates voice support status on LTE. – 0x00 - Voice is not supported – 0x01 - Voice is supported
pGSMCipher-DomainSysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
pWCDMA-CipherDomain-SysInfo	• Ciphering on the service domain. – 0x00 - No service – 0x01 - Circuit-switched only – 0x02 - Packet-switched only – 0x03 - Circuit-switched and packet-switched
pSysInfoNo-Change	• System Info No Change. • Flag used to notify clients that a request to select a network ended with no change in the PLMN. – 0x01 - No change in system information

8.591.2 Field Documentation

8.591.2.1 nas_AddCDMASysInfo* unpack_nas_SLQSSysInfoCallback_ind_t::pAddCDMASysInfo

8.591.2.2 nas_AddSysInfo* unpack_nas_SLQSSysInfoCallback_ind_t::pAddGSMSysInfo

8.591.2.3 uint16_t* unpack_nas_SLQSSysInfoCallback_ind_t::pAddHDRSysInfo

8.591.2.4 uint16_t* unpack_nas_SLQSSysInfoCallback_ind_t::pAddLTESysInfo

8.591.2.5 nas_AddSysInfo* unpack_nas_SLQSSysInfoCallback_ind_t::pAddWCDMASysInfo

8.591.2.6 nas_SrvStatusInfo* unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASrvStatusInfo

8.591.2.7 **nas_CDMASysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASysInfo

8.591.2.8 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCallBarringSysInfo

8.591.2.9 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCipherDomainSysInfo

8.591.2.10 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSrvStatusInfo

8.591.2.11 **nas_GSMSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSysInfo

8.591.2.12 **nas_SrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSrvStatusInfo

8.591.2.13 **nas_HDRSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSysInfo

8.591.2.14 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTESrvStatusInfo

8.591.2.15 **nas_LTESysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTESysInfo

8.591.2.16 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTEVoiceSupportSysInfo

8.591.2.17 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pSysInfoNoChange

8.591.2.18 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACallBarringSysInfo

8.591.2.19 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACipherDomainSysInfo

8.591.2.20 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASrvStatusInfo

8.591.2.21 **nas_WCDMASysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASysInfo

8.592 unpack_omaDmConfigTlv_t Struct Reference

Data Fields

- uint8_t state
- uint8_t userInputReq
- uint16_t userInputTimeout
- uint16_t alertmsglength
- uint8_t alertmsg [256]

8.592.1 Detailed Description

This structure will hold the SwiOmaDmConfig session parameters information.

Parameters

state	• 0x01 - OMA-DM Read Request • 0x02 - OMA-DM Change Request • 0x03 - OMA-DM Config Complete
userInputReq	- Bit mask of available user inputs • 0x00 - No user input required.Informational indication • 0x01 - Accept • 0x02 - Reject

userInput-Timeout	• Timeout for user input in minutes. A value of 0 means no time-out
alertmsglength	• Length of Alert message string in bytes
alertmsg	• Alert message in UCS2 (Max 256 characters)

8.592.2 Field Documentation

8.592.2.1 uint8_t unpack_omaDmConfigTlv_t::alertmsg[256]

8.592.2.2 uint16_t unpack_omaDmConfigTlv_t::alertmsglength

8.592.2.3 uint8_t unpack_omaDmConfigTlv_t::state

8.592.2.4 uint8_t unpack_omaDmConfigTlv_t::userInputReq

8.592.2.5 uint16_t unpack_omaDmConfigTlv_t::userInputTimeout

8.593 unpack_omaDmFotaTlv_t Struct Reference

Data Fields

- uint8_t state
- uint8_t userInputReq
- uint16_t userInputTimeout
- uint32_t fwdloadsize
- uint32_t fwloadComplete
- uint16_t updateCompleteStatus
- uint8_t severity
- uint16_t versionlength
- uint8_t version [256]
- uint16_t namelength
- uint8_t package_name [256]
- uint16_t descriptionlength
- uint8_t description [256]
- uint8_t sessionType

8.593.1 Detailed Description

This structure will hold the SwiOmaDmFota session parameters information.

Parameters

state	• 0x01 - No Firmware available • 0x02 - Query Firmware Download • 0x03 - Firmware Downloading • 0x04 - Firmware downloaded • 0x05 - Query Firmware Update • 0x06 - Firmware updating • 0x07 - Firmware updated
userInputReq	- Bit mask of available user inputs • 0x00 - No user input required.Informational indication • 0x01 - Accept • 0x02 - Reject
userInput-Timeout	• Timeout for user input in minutes. A value of 0 means no time-out
fwdloadsize	• The size (in bytes) of the firmware update package
fwloadComplete	• The number of bytes downloaded. Need to determine how often to send this message for progress bar notification. Every 500ms or 5% increment.
update-CompleteStatus	• This field should be looked at only when the OMADM session is complete. • See qaGobiApiTableSwiOMADMUpdateCompleteStatus.h for update complete status.
severity	• 0x01 - Mandatory • 0x02 - Optional
versionlength	• Length of FW Version string in bytes
version	• FW Version string in ASCII (Max 256 characters)
namelength	• Length Package Name string in bytes
package_name	• Package Name in UCS2 (Max 256 characters)
descriptionlength	• Length of description in bytes
description	• Description of Update Package in USC2 (Max 256 characters)

sessionType	• 0x00 - Client initiated • 0x01 - Network initiated

8.593.2 Field Documentation

8.593.2.1 uint8_t unpack_omaDmFotaTlv_t::description[256]

8.593.2.2 uint16_t unpack_omaDmFotaTlv_t::descriptionlength

8.593.2.3 uint32_t unpack_omaDmFotaTlv_t::fwdloadsize

8.593.2.4 uint32_t unpack_omaDmFotaTlv_t::fwloadComplete

8.593.2.5 uint16_t unpack_omaDmFotaTlv_t::namelength

8.593.2.6 uint8_t unpack_omaDmFotaTlv_t::package_name[256]

8.593.2.7 uint8_t unpack_omaDmFotaTlv_t::sessionType

8.593.2.8 uint8_t unpack_omaDmFotaTlv_t::severity

8.593.2.9 uint8_t unpack_omaDmFotaTlv_t::state

8.593.2.10 uint16_t unpack_omaDmFotaTlv_t::updateCompleteStatus

8.593.2.11 uint8_t unpack_omaDmFotaTlv_t::userInputReq

8.593.2.12 uint16_t unpack_omaDmFotaTlv_t::userInputTimeout

8.593.2.13 uint8_t unpack_omaDmFotaTlv_t::version[256]

8.593.2.14 uint16_t unpack_omaDmFotaTlv_t::versionlength

8.594 unpack_omaDmNotificationsTlv_t Struct Reference

Data Fields

- uint8_t notification
- uint16_t sessionStatus

8.594.1 Detailed Description

This structure will hold the SwiOmaDmConfig session notification parameters information.

Parameters

notification	• 0x01 - GPS settings change • 0x02 - Device reset • 0x03 - Device factory reset • 0x04 - CI-DC Session start • 0x05 - CI-DC Session end • 0x06 - CI-PRL Session start • 0x07 - CI-PRL Session end • 0x08 - CI-FUMO Session start • 0x09 - CI-FUMO session end • 0x0A - HFA-DC Session start • 0x0B - HFA-DC Session end • 0x0C - HFA-PRL Session start • 0x0D - HFA-PRL Session end • 0x0E - HFA-FUMO Session start • 0x0F - HFA-FUMO session end • 0x10 - NI Session start • 0x11 - NI session end
sessionStatus	• This field will set to the session status for notifications that occur at the end of a session, zero for all other notifications • See qaGobiApiTableSwiOMADMSessionStatus.h for session completion code.

8.594.2 Field Documentation

8.594.2.1 uint8_t unpack_omaDmNotificationsTlv_t::notification

8.594.2.2 uint16_t unpack_omaDmNotificationsTlv_t::sessionStatus

8.595 unpack_pds_GetPDSDefaults_t Struct Reference

Data Fields

- uint32_t * pOperation
- uint8_t * pTimeout
- uint32_t * pInterval
- uint32_t * pAccuracy

8.595.1 Detailed Description

Structure contain session configuration parameters.

Parameters

pOperation	• Current session operating mode – 0 - Standalone – 1 - MS based – 2 - MS assisted

pTimeout	• Maximum amount of time (seconds) to work on each fix, maximum is 255
pInterval	• Interval (seconds) between fix requests
pAccuracy	• Preferred accuracy threshold (meters)

8.595.2 Field Documentation

8.595.2.1 uint32_t* unpack_pds_GetPDSDefaults_t::pAccuracy

8.595.2.2 uint32_t* unpack_pds_GetPDSDefaults_t::pInterval

8.595.2.3 uint32_t* unpack_pds_GetPDSDefaults_t::pOperation

8.595.2.4 uint8_t* unpack_pds_GetPDSDefaults_t::pTimeout

8.596 unpack_pds_GetPDSState_t Struct Reference

Data Fields

- uint32_t * pEnabledStatus
- uint32_t * pTrackingStatus

8.596.1 Detailed Description

Structure contain parameters of current PDS state.

Parameters

pEnabledStatus	• Current PDS state – 0 - disable – 1 - enable
pTrackingStatus	• Current PDS tracking session state • Values: – 0x00 - Unknown – 0x01 - Inactive – 0x02 - Active

8.596.2 Field Documentation

8.596.2.1 uint32_t* unpack_pds_GetPDSState_t::pEnabledStatus

8.596.2.2 uint32_t* unpack_pds_GetPDSState_t::pTrackingStatus

8.597 unpack_pds_GetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t * pbAuto

8.597.1 Detailed Description

Structure contain the parameter of automatic tracking configuration for the NMEA COM port.

Parameters

pbAuto	• Automatic tracking enabled for NMEA COM port – 0x00 - Disabled – 0x01 - Enabled

8.597.2 Field Documentation

8.597.2.1 uint32_t* unpack_pds_GetPortAutomaticTracking_t::pbAuto

8.598 unpack_pds_GetServiceAutomaticTracking_t Struct Reference

Data Fields

- uint32_t * pbAuto

8.598.1 Detailed Description

Structure contain the parameter for automatic tracking state.

Parameters

pbAuto	• Automatic tracking session started for service – 0x00 - Disabled – 0x01 - Enabled

8.598.2 Field Documentation

8.598.2.1 uint32_t* unpack_pds_GetServiceAutomaticTracking_t::pbAuto

8.599 unpack_pds_GetXTRAAutomaticDownload_t Struct Reference

Data Fields

- uint32_t * pbEnabled
- uint16_t * pInterval

8.599.1 Detailed Description

Structure contain the parameter of automatic tracking configuration for the NMEA COM port.

Parameters

pbEnabled	• Automatic XTRA download status – 0 - Disabled – 1 - Enabled
pInterval	• Interval (hours) between XTRA downloads

8.599.2 Field Documentation

8.599.2.1 uint32_t* unpack_pds_GetXTRAAutomaticDownload_t::pbEnabled

8.599.2.2 uint16_t* unpack_pds_GetXTRAAutomaticDownload_t::pInterval

8.600 unpack_pds_GetXTRANetwork_t Struct Reference

Data Fields

- uint32_t * pPreference

8.600.1 Detailed Description

Structure contain the parameter for XTRA WWAN network preference

Parameters

pPreference	• XTRA WWAN network preference – 0x00 - None (any available network) – 0x01 - Home-only, only when on home systems – 0x02 - Roam-only, only when on non-home systems

8.600.2 Field Documentation

8.600.2.1 uint32_t* unpack_pds_GetXTRANetwork_t::pPreference

8.601 unpack_pds_GetXTRAValidity_t Struct Reference

Data Fields

- uint16_t * pGPSWeek
- uint16_t * pGPSWeekOffset
- uint16_t * pDuration

8.601.1 Detailed Description

Structure contain the parameter for XTRA database validity period

Parameters

pGPSWeek	• Starting GPS week of validity period
pGPSWeek-Offset	• Starting GPS week offset (minutes) of validity period
pDuration	• Length of validity period (hours)

8.601.2 Field Documentation

8.601.2.1 uint16_t* unpack_pds_GetXTRAValidity_t::pDuration

8.601.2.2 uint16_t* unpack_pds_GetXTRAValidity_t::pGPSWeek

8.601.2.3 uint16_t* unpack_pds_GetXTRAValidity_t::pGPSWeekOffset

8.602 unpack_pds_SetEventReport_Ind_t Struct Reference

Data Fields

- uint8_t has_PositionDataNMEA
- char PositionDataNMEA [200]
- uint8_t has_dLongitude
- double dLongitude
- uint8_t has_dLatitude
- double dLatitude
- uint8_t has_SessionStatus
- uint8_t SessionStatus
- uint8_t has_posSrc
- uint32_t posSrc

8.602.1 Detailed Description

Structure contain the parameter for the Set event report Indication. Please check has_<Param_Name> field for presence of optional parameters

Parameters

PositionDataNM-EA	• Position Data NMEA. • String containing the position data in NMEA sentence format (maximum 200 bytes).

dLongitude	• Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of the Greenwich Meridian. • Units - Decimal degrees • Range - -180 to +180 • Note - Value is in double float format
dLatitude	• Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. • Units: Decimal degrees • Range: -90 to +90 • Note - Value is in double float format
session_status	• Position Session Status • Status of current session. • Values – 0x00 - Success – 0x01 - In progress – 0x02 - General failure – 0x03 - Timeout – 0x04 - User ended the session – 0x05 - Bad parameter – 0x06 - Phone is offline – 0x07 - Engine is locked – 0x08 - E911 session in progress • Note - This TLV is always sent while a fix is being generated or if it halts due to an error. The Parsed Position Data TLV (0x13) is included if the session status is Success (0x00) or In Progress (0x01); otherwise this TLV is passed alone.
posSrc	• Position Source • This TLV is always sent with the Parsed Position Data TLV (0x13) when the latitude/longitude is marked as valid. Source of the position (bitmask). • Values – 0x00000001 - GPS – 0x00000002 – Cell ID – 0x00000004 – GLONASS – 0x00000008 – Network – 0x00000010 – External position injection

8.602.2 Field Documentation

8.602.2.1 double unpack_pds_SetEventReport_Ind_t::dLatitude

8.602.2.2 double unpack_pds_SetEventReport_Ind_t::dLongitude

8.602.2.3 uint8_t unpack_pds_SetEventReport_Ind_t::has_dLatitude

8.602.2.4 uint8_t unpack_pds_SetEventReport_Ind_t::has_dLongitude

8.602.2.5 uint8_t unpack_pds_SetEventReport_Ind_t::has_PositionDataNMEA

8.602.2.6 uint8_t unpack_pds_SetEventReport_Ind_t::has_posSrc

8.602.2.7 uint8_t unpack_pds_SetEventReport_Ind_t::has_SessionStatus

8.602.2.8 char unpack_pds_SetEventReport_Ind_t::PositionDataNMEA[200]

8.602.2.9 uint32_t unpack_pds_SetEventReport_Ind_t::posSrc

8.602.2.10 uint8_t unpack_pds_SetEventReport_Ind_t::SessionStatus

8.603 unpack_pds_SetPdsState_Ind_t Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t EnabledStatus
- uint32_t TrackingStatus

8.603.1 Detailed Description

Structure contain the parameter for the Set PDS State Indication.

Parameters

EnabledStatus	• GPS service state. • Values – 0x00 - Disable – 0x01 - Enable
TrackingStatus	• Tracking session state. • Values – 0x00 - Unknown – 0x01 - Inactive – 0x02 - Active

8.603.2 Field Documentation

8.603.2.1 uint32_t unpack_pds_SetPdsState_Ind_t::EnabledStatus

8.603.2.2 uint8_t unpack_pds_SetPdsState_Ind_t::TlvPresent

8.603.2.3 uint32_t unpack_pds_SetPdsState_Ind_t::TrackingStatus

8.604 unpack_pds_SLQSGetAGPSConfig_t Struct Reference

Data Fields

- uint32_t * pServerAddress
- uint32_t * pServerPort
- uint8_t * pServerURL
- uint8_t * pServerURLLength

8.604.1 Detailed Description

Structure contain the parameter for PDS AGPS (MS-based) configuration unpack.

Parameters

pServerAddress	• IPv4 address of AGPS server. "0" if not set
pServerPort	• Port number of AGPS server. "0" if not set
pServerURL	• URL of the AGPS server. "0" if not set
pServerURL-Length	• URL length of AGPS server. "0" if not set

8.604.2 Field Documentation

8.604.2.1 uint32_t* unpack_pds_SLQSGetAGPSConfig_t::pServerAddress

8.604.2.2 uint32_t* unpack_pds_SLQSGetAGPSConfig_t::pServerPort

8.604.2.3 uint8_t* unpack_pds_SLQSGetAGPSConfig_t::pServerURL

8.604.2.4 uint8_t* unpack_pds_SLQSGetAGPSConfig_t::pServerURLLength

8.605 unpack_pds_SLQSGetGPSStateInfo_t Struct Reference

Data Fields

- uint8_t EngineState
- uint32_t ValidMask
- uint64_t Latitude
- uint64_t Longitude
- uint32_t HorizontalUncertainty
- uint32_t Altitude
- uint32_t VerticalUncertainty
- uint32_t TimeStmp_tow_ms
- uint16_t TimeStmp_gps_week
- uint32_t Time_uncert_ms
- uint8_t Iono_valid
- uint32_t gps_ephemeris_sv_msk
- uint32_t gps_almanac_sv_msk
- uint32_t gps_health_sv_msk

- uint32_t gps_visible_sv_msk
- uint32_t glo_ephemeris_sv_msk
- uint32_t glo_almanac_sv_msk
- uint32_t glo_health_sv_msk
- uint32_t glo_visible_sv_msk
- uint32_t sbas_ephemeris_sv_msk
- uint32_t sbas_almanac_sv_msk
- uint32_t sbas_health_sv_msk
- uint32_t sbas_visible_sv_msk
- uint16_t xtra_start_gps_week
- uint16_t xtra_start_gps_minutes
- uint16_t xtra_valid_duration_hours

8.605.1 Detailed Description

Structure contain the parameter for GPS state Info.

Parameters

EngineState	• Values: – 0 - OFF – 1 - ON • This field is always valid
ValidMask	• Mask of valid state information data. • Values: – 0x00000001 - Position(latitude/longitude/horizontal uncertainty) – 0x00000002 - Altitude and vertical uncertainty – 0x00000004 - Time ms – 0x00000008 - Time week number – 0x00000010 - Time uncertainty – 0x00000020 - Iono validity – 0x00000040 - GPS ephemeris – 0x00000080 - GPS almanac – 0x00000100 - GPS health – 0x00000200 - GPS visible SVs – 0x00000400 - GLONASS ephemeris – 0x00000800 - GLONASS almanac – 0x00001000 - GLONASS health – 0x00002000 - GLONASS visible SVs – 0x00004000 - SBAS ephemeris – 0x00008000 - SBAS almanac – 0x00010000 - SBAS health – 0x00020000 - SBAS visible SVs – 0x00040000 - XTRA information

Latitude	• Latitude position referenced to the WGS-84 reference ellipsoid, counting positive angles north of the equator and negative angles south of the equator. • Units: Decimal degrees • Range: -90 to +90 degrees. • Value is in double float format (refer to IEEE Std 754-1985)
Longitude	• Longitude position referenced to the WGS-84 reference ellipsoid, counting positive angles east of the Greenwich Meridian and negative angles west of Greenwich meridian. • Units: Decimal degrees • Range: -180 to +180 degrees • Value is in double float format (refer to IEEE Std 754-1985)
Horizontal-Uncertainty	• Circular horizontal uncertainty (in meters). The uncertainty is provided at 63 percent confidence. • Value is in single float format (refer to IEEE Std 754-1985)
Altitude	• Height above the WGS-84 reference ellipsoid. Value conveys height (in meters) plus 500 m • Range -500 to 15883 • Value in single float format (refer to IEEE Std 754-1985)
Vertical-Uncertainty	• Vertical uncertainty (in meters). The uncertainty is provided at 68 percent confidence. • Value in single float format (refer to IEEE Std 754-1985)
TimeStmp_tow_ms	• Time stamp in GPS time of week(in milliseconds)
TimeStmp_gps_week	• GPS week number
Time_uncert_ms	• Time uncertainty (in milliseconds). The uncertainty is provided at 99 percent confidence.
Iono_valid	• Iono validity. • Values: – 0 - Invalid – 1 - Valid
gps_ephemeris_sv_msk	• GPS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available.
gps_almanac_sv_msk	• GPS SV mask for almanac; if the bit is set, almanac for that SV is available.
gps_health_sv_msk	• GPS SV mask for health; if the bit is set, health for that SV is available.

gps_visible_sv_msk	• GPS SV mask for visible Svs; if the bit is set, the SV is available.
glo_ephemeris_sv_msk	• GLONASS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available.
glo_almanac_sv_msk	• GLONASS SV mask for almanac; if the bit is set, almanac for that SV is available.
glo_health_sv_msk	• GLONASS SV mask for health; if the bit is set, health for that SV is available.
glo_visible_sv_msk	• GLONASS SV mask for visible SVs; if the bit is set, the SV is available.
sbas_ephemeris_sv_msk	• SBAS SV mask for ephemeris; if the bit is set, ephemeris for that SV is available.
sbas_almanac_sv_msk	• SBAS SV mask for almanac; if the bit is set, almanac for that SV is available.
sbas_health_sv_msk	• SBAS SV mask for health; if the bit is set, health for that SV is available.
sbas_visible_sv_msk	• SBAS SV mask for visible SVs; if the bit is set, the SV is available.
xtra_start_gps_week	• Current XTRA information is valid starting from this GPS week number
xtra_start_gps_minutes	• Current XTRA information is valid starting from the GPS minutes with the GPS week
xtra_valid_duration_hours	• XTRA information is valid for this many hours starting from the specified GPS week/minutes

8.605.2 Field Documentation

8.605.2.1 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::Altitude

8.605.2.2 uint8_t unpack_pds_SLQSGetGPSStateInfo_t::EngineState

8.605.2.3 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_almanac_sv_msk

8.605.2.4 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_ephemeris_sv_msk

8.605.2.5 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_health_sv_msk

8.605.2.6 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::glo_visible_sv_msk

8.605.2.7 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_almanac_sv_msk

8.605.2.8 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_ephemeris_sv_msk

8.605.2.9 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_health_sv_msk

8.605.2.10 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::gps_visible_sv_msk

8.605.2.11 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::HorizontalUncertainty

8.605.2.12 uint8_t unpack_pds_SLQSGetGPSStateInfo_t::Iono_valid

8.605.2.13 uint64_t unpack_pds_SLQSGetGPSStateInfo_t::Latitude

8.605.2.14 uint64_t unpack_pds_SLQSGetGPSStateInfo_t::Longitude

8.605.2.15 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_almanac_sv_msk

8.605.2.16 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_ephemeris_sv_msk

8.605.2.17 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_health_sv_msk

8.605.2.18 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::sbas_visible_sv_msk

8.605.2.19 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::Time_uncert_ms

8.605.2.20 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::TimeStmp_gps_week

8.605.2.21 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::TimeStmp_tow_ms

8.605.2.22 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::ValidMask

8.605.2.23 uint32_t unpack_pds_SLQSGetGPSStateInfo_t::VerticalUncertainty

8.605.2.24 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_start_gps_minutes

8.605.2.25 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_start_gps_week

8.605.2.26 uint16_t unpack_pds_SLQSGetGPSStateInfo_t::xtra_valid_duration_hours

8.606 unpack_qmi_t Struct Reference

Data Fields

- enum msgtype type
- uint16_t msgid
- uint16_t xid

8.606.1 Detailed Description

qmi response context

Parameters

out	*type*	message type
out	*msgid*	message id
out	*xid*	transaction id

8.606.2 Field Documentation

8.606.2.1 uint16_t unpack_qmi_t::msgid

8.606.2.2 enum msgtype unpack_qmi_t::type

8.606.2.3 uint16_t unpack_qmi_t::xid

8.607 unpack_qos_dataRate_t Struct Reference

Data Fields

- uint32_t dataRateMax
- uint32_t guaranteedRate

8.607.1 Detailed Description

This structure contains the IP flow data rate min max

Parameters

dataRateMax	Maximum required data rate (bits per second)
guaranteedRate	Minimum guaranteed data rate (bits per second)

8.607.2 Field Documentation

8.607.2.1 uint32_t unpack_qos_dataRate_t::dataRateMax

8.607.2.2 uint32_t unpack_qos_dataRate_t::guaranteedRate

8.608 unpack_qos_IPv4Addr_t Struct Reference

Data Fields

- uint32_t addr
- uint32_t subnetMask

8.608.1 Detailed Description

This structure contains the IPv4 filter address

Parameters

addr	IPv4 address
subnetMask	A packet matches if: • (addr and subnetMask) == (IP pkt addr & subnetMask) Callers to set up a filter with a range of source addresses, if needed; subnet mask of all 1s (255.255.255.255) specifies a single address value

8.608.2 Field Documentation

8.608.2.1 uint32_t unpack_qos_IPv4Addr_t::addr

8.608.2.2 uint32_t unpack_qos_IPv4Addr_t::subnetMask

8.609 unpack_qos_IPv6Addr_t Struct Reference

Data Fields

- uint8_t addr [16]
- uint8_t prefixLen

8.609.1 Detailed Description

This structure contains the IPv6 filter address

Parameters

addr	IPv6 address (in network byte order); this is a 16-byte byte array (in Big-endian format)
prefixLen	IPv6 filter prefix length; can take a value between 0 and 128 Note: A packet matches if the IPv6 source address bytes until the prefix lengths are equal. Therefore prefix length can be used to set a filter with a range of source addresses. A prefix length of 128 specifies a single address value.

8.609.2 Field Documentation

8.609.2.1 uint8_t unpack_qos_IPv6Addr_t::addr[16]

8.609.2.2 uint8_t unpack_qos_IPv6Addr_t::prefixLen

8.610 unpack_qos_IPv6TrafCls_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.610.1 Detailed Description

This structure contains the IPv6 filter traffic class

Parameters

val	The traffic class value
mask	The packet matches the traffic class filter if: (IPv6_filter_traffic_class_val and IPv6_filter_traffic_class_mask) == (Traffic class value in the IP packet & IPv6_filter_traffic_class_mask) Example: • IPv6_filter_tc_val = 00101000 • IPv6_filter_tc_mask = 11111100 Filter will compare only the first 6 bits in IPv6_filter_traffic_class with the first 6 bits in the traffic class field of the IP packet; first 6 bits in the traffic class field of the IP packet must be 001010 to match filter; last 2 bits can be anything, since they are ignored by filtering

8.610.2 Field Documentation

8.610.2.1 uint8_t unpack_qos_IPv6TrafCls_t::mask

8.610.2.2 uint8_t unpack_qos_IPv6TrafCls_t::val

8.611 unpack_qos_pktErrRate_t Struct Reference

Data Fields

- uint16_t multiplier
- uint16_t exponent

8.611.1 Detailed Description

This structure contains the IP flow packet error rate

Parameters

multiplier	Factor m in calculating packet error rate: E = m∗10∗∗(-p)
exponent	Factor p in calculating packet error rate (see above)

8.611.2 Field Documentation

8.611.2.1 uint16_t unpack_qos_pktErrRate_t::exponent

8.611.2.2 uint16_t unpack_qos_pktErrRate_t::multiplier

8.612 unpack_qos_Port_t Struct Reference

Data Fields

- uint16_t port
- uint16_t range

8.612.1 Detailed Description

This structure contains the Port Filter

Parameters

port	port value of the filter
range	range specifies the number of ports to be included in the filter starting from port; filter will match if port in the IP packet lies between port and (port + range) Range value of 0 implies that only one value of the port is valid, as specified by the port

8.612.2 Field Documentation

8.612.2.1 uint16_t unpack_qos_Port_t::port

8.612.2.2 uint16_t unpack_qos_Port_t::range

8.613 unpack_qos_QosFlowInfo_t Struct Reference

Data Fields

- unpack_qos_QosFlowInfoState_t QFlowState
- uint8_t is_TxQFlowGranted_Available
- unpack_qos_swiQosFlow_t TxQFlowGranted
- uint8_t is_RxQFlowGranted_Available
- unpack_qos_swiQosFlow_t RxQFlowGranted
- uint8_t NumTxFilters
- unpack_qos_swiQosFilter_t TxQFilter [25]
- uint8_t NumRxFilters
- unpack_qos_swiQosFilter_t RxQFilter [25]
- uint8_t BearerID

8.613.1 Detailed Description

Structure with QoS flow details.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

QFlowState	• QoS flow state information, please check unpack_qos_QosFlowInfoState_t for more information
is_TxQFlowGranted_Available	• TRUE if optional TxQFlowGranted is available
TxQFlowGranted	• The Tx Qos flow granted, please check unpack_qos_swiQosFlow_t for more information
is_RxQFlowGranted_Available	• TRUE if optional RxQFlowGranted is available
RxQFlowGranted	• The Rx Qos flow granted, please check unpack_qos_swiQosFlow_t for more information
NumTxFilters	• Number of Tx filters available
TxQFilter	• The Tx Qos filter, please check unpack_qos_swiQosFilter_t for more information • See LITEQMI_MAX_QOS_FILTERS for more information
NumRxFilters	• Number of Tx filters available
RxQFilter	• The Rx Qos filter, please check unpack_qos_swiQosFilter_t for more information • See LITEQMI_MAX_QOS_FILTERS for more information
BearerID	• The bearer ID • Bearer ID or Radio Link Protocol (RLP) ID of the activated flow. • Valid Values - 0 to 16 • 0xFF - Invalid value.

8.613.2 Field Documentation

8.613.2.1 uint8_t unpack_qos_QosFlowInfo_t::BearerID

8.613.2.2 uint8_t unpack_qos_QosFlowInfo_t::is_RxQFlowGranted_Available

8.613.2.3 uint8_t unpack_qos_QosFlowInfo_t::is_TxQFlowGranted_Available

8.613.2.4 uint8_t unpack_qos_QosFlowInfo_t::NumRxFilters

8.613.2.5 uint8_t unpack_qos_QosFlowInfo_t::NumTxFilters

8.613.2.6 unpack_qos_QosFlowInfoState_t unpack_qos_QosFlowInfo_t::QFlowState

8.613.2.7 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::RxQFilter[25]

8.613.2.8 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::RxQFlowGranted

8.613.2.9 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::TxQFilter[25]

8.613.2.10 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::TxQFlowGranted

8.614 unpack_qos_QosFlowInfoState_t Struct Reference

Data Fields

- uint32_t id
- uint8_t isNewFlow
- uint8_t state

8.614.1 Detailed Description

This structure contains QoS flow state

Parameters

id	QoS identifier
isNewFlow	• 1 – Newly added flow • 0 – Existing flow
state	This indicates that the flow that was added/modified/deleted: • 0x01 – Flow activated • 0x02 – Flow modified • 0x03 – Flow deleted • 0x04 – Flow suspended • 0x05 – Flow enabled • 0x06 – Flow disabled

8.614.2 Field Documentation

8.614.2.1 uint32_t unpack_qos_QosFlowInfoState_t::id

8.614.2.2 uint8_t unpack_qos_QosFlowInfoState_t::isNewFlow

8.614.2.3 uint8_t unpack_qos_QosFlowInfoState_t::state

8.615 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference

Data Fields

- uint8_t NWQoSStatus

8.615.1 Detailed Description

Structure that contains the response to get NW QoS status command

Parameters

NWQoSStatus	Network QoS support status • 0 – No QoS support in network • 1 – Network supports QoS

8.615.2 Field Documentation

8.615.2.1 uint8_t unpack_qos_SLQSQosGetNetworkStatus_t::NWQoSStatus

8.616 unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference

Data Fields

- uint32_t apnId
- uint8_t ambr_ul
- uint8_t ambr_dl
- uint8_t ambr_ul_ext
- uint8_t ambr_dl_ext
- uint8_t ambr_ul_ext2
- uint8_t ambr_dl_ext2

8.616.1 Detailed Description

Structure that contains extra APN parameters

Parameters

apnId	• APN id • ID identifing the APN that the client would like to query the AMBR params
ambr_ul	• APN AMBR uplink • APN AMBR uplink values from 1 kbps to 8640 kbps

ambr_dl	• APN AMBR downlink • APN AMBR downlink values from 1 kbps to 8640 kbps
ambr_ul_ext	• Extended APN AMBR uplink • APN AMBR uplink values from 8700 kbps to 256 Mbps
ambr_dl_ext	• Extended APN AMBR downlink • APN AMBR downlink values from 8700 kbps to 256 Mbps
ambr_ul_ext2	• Second extended APN AMBR uplink • APN AMBR uplink values from 256 Mbps to 65280 Mbps
ambr_dl_ext2	• Second extended APN AMBR downlink • APN AMBR downlink values from 256 Mbps to 65280 Mbps

8.616.2 Field Documentation

8.616.2.1 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl

8.616.2.2 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext

8.616.2.3 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext2

8.616.2.4 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul

8.616.2.5 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext

8.616.2.6 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext2

8.616.2.7 uint32_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.617 unpack_qos_SLQSQosSwiReadDataStats_t Struct Reference

Data Fields

- uint32_t apnId
- uint32_t total_tx_pkt
- uint32_t total_tx_pkt_drp
- uint32_t total_rx_pkt
- uint64_t total_tx_bytes
- uint64_t total_tx_bytes_drp
- uint64_t total_rx_bytes
- uint32_t numQosFlow
- unpack_QosFlowStat_t qosFlow [10]

8.617.1 Detailed Description

Structure that contains APN data statistics

Parameters

apnId	• APN id • ID identifing the connected APN that the client would like to query the data statistic for
total_tx_pkt	• sum of all packets sent
total_tx_pkt_drp	• sum of all(TX) packets dropped
total_rx_pkt	• sum of all packets received
total_tx_bytes	• sum of all bytes sent
total_tx_bytes_drp	• sum of all(TX) bytes dropped
total_rx_bytes	• number of received bytes for the QoS flow ID
numQosFlow	• pointer to number of QoS flow Stat
qosFlow[LITEQMI_MAX_QOS_FLOW_PER_APN_STATS]	• Data statistic per QoS flow • See unpack_QosFlowStat_t for more information • See LITEQMI_MAX_QOS_FLOW_PER_APN_STATS for more information

8.617.2 Field Documentation

8.617.2.1 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::apnId

8.617.2.2 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::numQosFlow

8.617.2.3 unpack_QosFlowStat_t unpack_qos_SLQSQosSwiReadDataStats_t::qosFlow[10]

8.617.2.4 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_bytes

8.617.2.5 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_pkt

8.617.2.6 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes

8.617.2.7 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes_drp

8.617.2.8 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt

8.617.2.9 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt_drp

8.618 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference

Data Fields

- uint8_t NumFlows
- unpack_qos_QosFlowInfo_t QosFlowInfo [8]

8.618.1 Detailed Description

Structure with QoS event details

Parameters

NumFlows	• Number of QoS flows available
QosFlowInfo	• The Qos flow details, please check unpack_qos_QosFlowInfo_t for more information • See LITEQMI_MAX_QOS_FLOWS for more information

8.618.2 Field Documentation

8.618.2.1 uint8_t unpack_qos_SLQSSetQosEventCallback_ind_t::NumFlows

8.618.2.2 unpack_qos_QosFlowInfo_t unpack_qos_SLQSSetQosEventCallback_ind_t::QosFlowInfo[8]

8.619 unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference

Data Fields

- uint8_t status

8.619.1 Detailed Description

Structure with network's QoS status

Parameters

status	Network QoS support status • 0x00 – Current network does not support QoS • 0x01 – Current network supports QoS

Note

- Technology Supported: CDMA

8.619.2 Field Documentation

8.619.2.1 uint8_t unpack_qos_SLQSSetQosNWStatusCallback_ind_t::status

8.620 unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference

Data Fields

- uint16_t event

8.620.1 Detailed Description

Structure with QoS primary flow events

Parameters

event	Event which causes this indication: • 0x0001 – Primary flow QoS modify operation success • 0x0002 – Primary flow QoS modify operation failure

8.620.2 Field Documentation

8.620.2.1 uint16_t unpack_qos_SLQSSetQosPriEventCallback_ind_t::event

8.621 unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference

Data Fields

- uint32_t id
- uint8_t status
- uint8_t event
- uint8_t reason

8.621.1 Detailed Description

Structure with QoS status indication details

Parameters

id	• Index identifying the QoS flow whose status is being reported
status	Current QoS flow status: • 0x01 – QMI_QOS_STATUS_ACTIVATED • 0x02 – QMI_QOS_STATUS_SUSPENDED • 0x03 – QMI_QOS_STATUS_GONE

event	• 0x01 – QMI_QOS_ACTIVATED_EV • 0x02 – QMI_QOS_SUSPENDED_EV • 0x03 – QMI_QOS_GONE_EV • 0x04 – QMI_QOS_MODIFY_ACCEPTED_EV • 0x05 – QMI_QOS_MODIFY_REJECTED_EV • 0x06 – QMI_QOS_INFO_CODE_UPDATED_EV
reason	• 0x01 - QMI_QOS_INVALID_PARAMS • 0x02 - QMI_QOS_INTERNAL_CALL_ENDED • 0x03 - QMI_QOS_INTERNAL_ERROR • 0x04 - QMI_QOS_INSUFFICIENT_LOCAL_Resources • 0x05 - QMI_QOS_TIMED_OUT_OPERATION • 0x06 - QMI_QOS_INTERNAL_UNKNOWN_CAUSE_CODE • 0x07 - QMI_QOS_INTERNAL_MODIFY_IN_PROGRESS • 0x08 - QMI_QOS_NOT_SUPPORTED • 0x09 - QMI_QOS_NOT_AVAILABLE • 0x0A - QMI_QOS_NOT_GUARANTEED • 0x0B - QMI_QOS_INSUFFICIENT_NETWORK_RESOURCES • 0x0C - QMI_QOS_AWARE_SYSTEM • 0x0D - QMI_QOS_UNAWARE_SYSTEM • 0x0E - QOS_REJECTED_OPERATION • 0x0F - QMI_QOS_WILL_GRANT_WHEN_QOS_RESUMED • 0x10 - QMI_QOS_NETWORK_CALL_ENDED • 0x11 - QMI_QOS_NETWORK_SERVICE_NOT_AVAILABLE • 0x12 - QMI_QOS_NETWORK_L2_LINK_RELEASED • 0x13 - QMI_QOS_NETWORK_L2_LINK_REESTAB_REJ • 0x14 - QMI_QOS_NETWORK_L2_LINK_REESTAB_IND • 0x15 - QMI_QOS_NETWORK_UNKNOWN_CAUSE_CODE • 0x16 - QMI_NETWORK_BUSY

8.621.2 Field Documentation

8.621.2.1 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::event

8.621.2.2 uint32_t unpack_qos_SLQSSetQosStatusCallback_ind_t::id

8.621.2.3 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::reason

8.621.2.4 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::status

8.622 unpack_qos_swiQosFilter_t Struct Reference

Data Fields

- uint8_t index
- uint8_t version
- uint8_t is_IPv4SrcAddr_Available

- unpack_qos_IPv4Addr_t IPv4SrcAddr
- uint8_t is_IPv4DstAddr_Available
- unpack_qos_IPv4Addr_t IPv4DstAddr
- uint8_t is_NxtHdrProto_Available
- uint8_t NxtHdrProto
- uint8_t is_IPv4Tos_Available
- unpack_qos_Tos_t IPv4Tos
- uint8_t is_IPv6SrcAddr_Available
- unpack_qos_IPv6Addr_t IPv6SrcAddr
- uint8_t is_IPv6DstAddr_Available
- unpack_qos_IPv6Addr_t IPv6DstAddr
- uint8_t is_IPv6TrafCls_Available
- unpack_qos_IPv6TrafCls_t IPv6TrafCls
- uint8_t is_IPv6Label_Available
- uint32_t IPv6Label
- uint8_t is_TCPSrcPort_Available
- unpack_qos_Port_t TCPSrcPort
- uint8_t is_TCPDstPort_Available
- unpack_qos_Port_t TCPDstPort
- uint8_t is_UDPSrcPort_Available
- unpack_qos_Port_t UDPSrcPort
- uint8_t is_UDPDstPort_Available
- unpack_qos_Port_t UDPDstPort
- uint8_t is_EspSpi_Available
- uint32_t EspSpi
- uint8_t is_Precedence_Available
- uint16_t Precedence
- uint8_t is_Id_Available
- uint16_t Id
- uint8_t is_TranSrcPort_Available
- unpack_qos_Port_t TranSrcPort
- uint8_t is_TranDstPort_Available
- unpack_qos_Port_t TranDstPort

8.622.1 Detailed Description

This structure contains the QoS Filter Request.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

index	Mandatory parameter IP filter index Integer that uniquely identifies each filter instance This TLV must be present in the request
version	Mandatory parameter IP filter version Identifies whether the filter is associated with IPv4 or IPv6; value specified also implies that only TLVs defined for that IP version, i.e., TLVs with IPv4 or IPv6 in the name, can be specified • 0x04 – IPv4 • 0x06 – Ipv6
IPv4SrcAddr	IPv4 filter soruce address See unpack_qos_IPv4Addr_t for more information • Implemented only for unsolicited indication
IPv4DstAddr	IPv4 filter destination address See unpack_qos_IPv4Addr_t for more information • Implemented only for unsolicited indication

NxtHdrProto	IP filter next header protocol This TLV must be present if any non-IP filter TLV(s) are provided If this field is specified, only IP packets belonging to specified higher layer protocol are considered when filtering The following protocols may be specified: • 0x01 = ICMP • 0x06 = TCP • 0x11 = UDP • 0x32 = ESP Note: The next header protocol field will be set to 0xFD (TCP & UDP) if a TFT is received specifying a source or destination port number, but IP next header type is not specified.
IPv4Tos	IPv4 filter type of service See unpack_qos_Tos_t for more information
IPv6SrcAddr	IPv6 filter soruce address See unpack_qos_IPv6Addr_t for more information • Implemented only for unsolicited indication
IPv6DstAddr	IPv6 filter destination address See unpack_qos_IPv6Addr_t for more information • Implemented only for unsolicited indication
IPv6TrafCls	IPv6 filter traffic class See unpack_qos_IPv6TrafCls_t for more information
IPv6Label	IPv6 flow label Packet matches the IPv6 flow label filter if: (∗pIPv6Label == flow label in the IPv6 header) • Implemented only for unsolicited indication
TCPSrcPort	TCP filter source port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
TCPDstPort	TCP filter destination port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
UDPSrcPort	UDP filter source port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
UDPDstPort	UDP filter destination port filter See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
EspSpi	ESP filter security policy index Security policy index to uniquely identify each IP flow for filtering encrypted packets for encapsulating security payload • Implemented only for unsolicited indication
Precedence	Filter Precedence Specifies the order in which filters are applied; lower numerical value has higher precedence Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored
Id	Filter ID Unique identifier for each filter;filter ID is assigned by the modem Note: This TLV only applies to network-initiated QoS; QoS requests containing this TLV from control points will be ignored
TranSrcPort	Transport protocolfilter source port See unpack_qos_Port_t for more information • Implemented only for unsolicited indication
UDPDstPort	Transport protocol filter destination port See unpack_qos_Port_t for more information • Implemented only for unsolicited indication

8.622.2 Field Documentation

8.622.2.1 uint32_t unpack_qos_swiQosFilter_t::EspSpi

8.622.2.2 uint16_t unpack_qos_swiQosFilter_t::Id

8.622.2.3 uint8_t unpack_qos_swiQosFilter_t::index

8.622.2.4 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4DstAddr

8.622.2.5 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4SrcAddr

8.622.2.6 unpack_qos_Tos_t unpack_qos_swiQosFilter_t::IPv4Tos

8.622.2.7 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6DstAddr

8.622.2.8 uint32_t unpack_qos_swiQosFilter_t::IPv6Label

8.622.2.9 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6SrcAddr

8.622.2.10 unpack_qos_IPv6TrafCls_t unpack_qos_swiQosFilter_t::IPv6TrafCls

8.622.2.11 uint8_t unpack_qos_swiQosFilter_t::is_EspSpi_Available

8.622.2.12 uint8_t unpack_qos_swiQosFilter_t::is_Id_Available

8.622.2.13 uint8_t unpack_qos_swiQosFilter_t::is_IPv4DstAddr_Available

8.622.2.14 uint8_t unpack_qos_swiQosFilter_t::is_IPv4SrcAddr_Available

8.622.2.15 uint8_t unpack_qos_swiQosFilter_t::is_IPv4Tos_Available

8.622.2.16 uint8_t unpack_qos_swiQosFilter_t::is_IPv6DstAddr_Available

8.622.2.17 uint8_t unpack_qos_swiQosFilter_t::is_IPv6Label_Available

8.622.2.18 uint8_t unpack_qos_swiQosFilter_t::is_IPv6SrcAddr_Available

8.622.2.19 uint8_t unpack_qos_swiQosFilter_t::is_IPv6TrafCls_Available

8.622.2.20 uint8_t unpack_qos_swiQosFilter_t::is_NxtHdrProto_Available

8.622.2.21 uint8_t unpack_qos_swiQosFilter_t::is_Precedence_Available

8.622.2.22 uint8_t unpack_qos_swiQosFilter_t::is_TCPDstPort_Available

8.622.2.23 uint8_t unpack_qos_swiQosFilter_t::is_TCPSrcPort_Available

8.622.2.24 uint8_t unpack_qos_swiQosFilter_t::is_TranDstPort_Available

8.622.2.25 uint8_t unpack_qos_swiQosFilter_t::is_TranSrcPort_Available

8.622.2.26 uint8_t unpack_qos_swiQosFilter_t::is_UDPDstPort_Available

8.622.2.27 uint8_t unpack_qos_swiQosFilter_t::is_UDPSrcPort_Available

8.622.2.28 uint8_t unpack_qos_swiQosFilter_t::NxtHdrProto

8.622.2.29 uint16_t unpack_qos_swiQosFilter_t::Precedence

8.622.2.30 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPDstPort

8.622.2.31 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPSrcPort

8.622.2.32 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranDstPort

8.622.2.33 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranSrcPort

8.622.2.34 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPDstPort

8.622.2.35 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPSrcPort

8.622.2.36 uint8_t unpack_qos_swiQosFilter_t::version

8.623 unpack_qos_swiQosFlow_t Struct Reference

Data Fields

- uint8_t index
- uint8_t is_ProfileId3GPP2_Available
- uint16_t ProfileId3GPP2
- uint8_t is_val_3GPP2Pri_Available
- uint8_t val_3GPP2Pri
- uint8_t is_TrafficClass_Available
- uint8_t TrafficClass
- uint8_t is_DataRate_Available
- unpack_qos_dataRate_t DataRate
- uint8_t is_TokenBucket_Available
- unpack_qos_tokenBucket_t TokenBucket
- uint8_t is_Latency_Available
- uint32_t Latency
- uint8_t is_Jitter_Available
- uint32_t Jitter
- uint8_t is_PktErrRate_Available
- unpack_qos_pktErrRate_t PktErrRate
- uint8_t is_MinPolicedPktSz_Available
- uint32_t MinPolicedPktSz
- uint8_t is_MaxAllowedPktSz_Available
- uint32_t MaxAllowedPktSz
- uint8_t is_val_3GPPResResidualBER_Available
- uint16_t val_3GPPResResidualBER
- uint8_t is_val_3GPPTraHdlPri_Available
- uint8_t val_3GPPTraHdlPri
- uint8_t is_val_3GPPImCn_Available
- uint8_t val_3GPPImCn
- uint8_t is_val_3GPPSigInd_Available
- uint8_t val_3GPPSigInd
- uint8_t is_LteQci_Available
- uint8_t LteQci

8.623.1 Detailed Description

This structure contains the QoS Flow Request.

Please check is_<Param_Name>_Available field for presence of optional parameters

Parameters

index	• Mandatory parameter • IP flow index • Integer that uniquely identifies each flow instance • Unique index must be assigned by the control point to every flow_spec instance
ProfileId3GPP2	• IP flow 3GPP2 profile ID • A profile ID is shorthand for a defined set of QoS flow parameters specified by the network; to be present while requesting QoS for a CDMA device
val_3GPP2Pri	• IP flow 3GPP2 flow priority • Flow priority used by the network in case of contention between flows with same QoS; this parameter applies for CDMA devices
TrafficClass	• IP flow traffic class • Integer that designates the requested traffic class: • 0 – Conversational • 1 – Streaming • 2 – Interactive • 3 – Background
DataRate	• IP flow data rate min max • See unpack_qos_dataRate_t for more information
TokenBucket	• IP flow data rate token bucket • See unpack_qos_tokenBucket_t for more information
Latency	• IP flow latency • Maximum delay (in milliseconds) that can be tolerated by an IP packet during transfer through the wireless link
Jitter	• IP flow jitter • Difference between the maximum and minimum latency (in milliseconds) that can be tolerated by an IP packet during the transfer through the wireless link
PktErrRate	• IP flow packet error rate • See unpack_qos_pktErrRate_t for more information

MinPolicedPktSz	• IP flow minimum policed packet size • Integer that defines the minimum packet size (in bytes) that will be policed for QoS guarantees; any IP packets that are smaller than the minimum specified policed size may not receive requested QoS
MaxAllowedPktSz	• IP flow maximum allowed packet size • Integer that defines the maximum packet size (in bytes) allowed in the IP flow; any IP packets greater in size than the maximum allowed packet size are not queued for transmission
val_3GPPResResidualBER	• IP flow 3GPP residual bit error rate • residual_bit_error_rate • 0 = 5∗10-2 residual BER • 1 = 1∗10-2 residual BER • 2 = 5∗10-3 residual BER • 3 = 4∗10-3 residual BER • 4 = 1∗10-3 residual BER • 5 = 1∗10-4 residual BER • 6 = 1∗10-5 residual BER • 7 = 1∗10-6 residual BER • 8 = 6∗10-8 residual BER • Integer that indicates the undetected BER for each IP flow in the delivered packets; Tapplies only to 3GPP networks
val_3GPPTraHdlPri	• 3GPP traffic handling priority • 0 – Relative traffic handling priority 1 • 1 – Relative traffic handling priority 2 • 2 – Relative traffic handling priority 3 • Defines the relative priority of the flow; applies only to 3GPP networks
val_3GPPImCn	• IP flow 3GPP IM CN flag • IM CN subsystem signaling flag: • 0x00 – FALSE • 0x01 – TRUE • This parameter applies only to 3GPP networks
val_3GPPSigInd	• IP flow 3GPP signaling indication • 0x00 – FALSE • 0x01 – TRUE • This parameter applies only to 3GPP networks

LteQci	• LTE QoS Class Identifier • QoS Class Identifier(QCI) is a required parameter to request QoS in LTE • QCI values: – QCI value 0 requests the network to assign the appropriate QCI value – QCI values 1-4 are associated with guaranteed bitrates – QCI values 5-9 are associated with nonguaranteed bitrates, so the values specified as guaranteed and maximum bitrates are ignored

8.623.2 Field Documentation

8.623.2.1 unpack_qos_dataRate_t unpack_qos_swiQosFlow_t::DataRate

8.623.2.2 uint8_t unpack_qos_swiQosFlow_t::index

8.623.2.3 uint8_t unpack_qos_swiQosFlow_t::is_DataRate_Available

8.623.2.4 uint8_t unpack_qos_swiQosFlow_t::is_Jitter_Available

8.623.2.5 uint8_t unpack_qos_swiQosFlow_t::is_Latency_Available

8.623.2.6 uint8_t unpack_qos_swiQosFlow_t::is_LteQci_Available

8.623.2.7 uint8_t unpack_qos_swiQosFlow_t::is_MaxAllowedPktSz_Available

8.623.2.8 uint8_t unpack_qos_swiQosFlow_t::is_MinPolicedPktSz_Available

8.623.2.9 uint8_t unpack_qos_swiQosFlow_t::is_PktErrRate_Available

8.623.2.10 uint8_t unpack_qos_swiQosFlow_t::is_ProfileId3GPP2_Available

8.623.2.11 uint8_t unpack_qos_swiQosFlow_t::is_TokenBucket_Available

8.623.2.12 uint8_t unpack_qos_swiQosFlow_t::is_TrafficClass_Available

8.623.2.13 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPP2Pri_Available

8.623.2.14 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPImCn_Available

8.623.2.15 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPResResidualBER_Available

8.623.2.16 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPSigInd_Available

8.623.2.17 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPTraHdlPri_Available

8.623.2.18 uint32_t unpack_qos_swiQosFlow_t::Jitter

8.623.2.19 uint32_t unpack_qos_swiQosFlow_t::Latency

8.623.2.20 uint8_t unpack_qos_swiQosFlow_t::LteQci

8.623.2.21 uint32_t unpack_qos_swiQosFlow_t::MaxAllowedPktSz

8.623.2.22 uint32_t unpack_qos_swiQosFlow_t::MinPolicedPktSz

8.623.2.23 unpack_qos_pktErrRate_t unpack_qos_swiQosFlow_t::PktErrRate

8.623.2.24 uint16_t unpack_qos_swiQosFlow_t::ProfileId3GPP2

8.623.2.25 unpack_qos_tokenBucket_t unpack_qos_swiQosFlow_t::TokenBucket

8.623.2.26 uint8_t unpack_qos_swiQosFlow_t::TrafficClass

8.623.2.27 uint8_t unpack_qos_swiQosFlow_t::val_3GPP2Pri

8.623.2.28 uint8_t unpack_qos_swiQosFlow_t::val_3GPPImCn

8.623.2.29 uint16_t unpack_qos_swiQosFlow_t::val_3GPPResResidualBER

8.623.2.30 uint8_t unpack_qos_swiQosFlow_t::val_3GPPSigInd

8.623.2.31 uint8_t unpack_qos_swiQosFlow_t::val_3GPPTraHdlPri

8.624 unpack_qos_tokenBucket_t Struct Reference

Data Fields

- uint32_t peakRate
- uint32_t tokenRate
- uint32_t bucketSz

8.624.1 Detailed Description

This structure contains the TP flow data rate token bucket

Parameters

peakRate	Maximum rate at which data can be transmitted when the token bucket is full (bits per second)
tokenRate	Rate at which tokens will be put in the token bucket (bits per second); a token is required to be present in the bucket to send a byte of data
bucketSz	Maximum number of tokens that can be accumulated at any instance (bytes); controls the size of the burst that is allowed at any given time

8.624.2 Field Documentation

8.624.2.1 uint32_t unpack_qos_tokenBucket_t::bucketSz

8.624.2.2 uint32_t unpack_qos_tokenBucket_t::peakRate

8.624.2.3 uint32_t unpack_qos_tokenBucket_t::tokenRate

8.625 unpack_qos_Tos_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.625.1 Detailed Description

This structure contains the IPv4 filter type of service

Parameters

val	Type of service value
mask	Packet matches the TOS filter if: (IPv4_filter_tos_val and IPv4_filter_tos_mask) == (TOS value in the IP packet & IPv4_filter_tos_mask) Example: • IPv4_filter_tos_val = 00101000 • IPv4_filter_tos_mask = 11111100 The filter will compare only the first 6 bits in the IPv4_filter_type_of_service with the first 6 bits in the TOS field of the IP packet. The first 6 bits in the TOS field of the IP packet must be 001010 to match the filter. The last 2 bits can be anything since they are ignored by filtering.

8.625.2 Field Documentation

8.625.2.1 uint8_t unpack_qos_Tos_t::mask

8.625.2.2 uint8_t unpack_qos_Tos_t::val

8.626 unpack_QosFlowStat_t Struct Reference

Data Fields

- uint32_t bearerId
- uint32_t tx_pkt
- uint32_t tx_pkt_drp
- uint64_t tx_bytes
- uint64_t tx_bytes_drp

8.626.1 Detailed Description

This structure contains the Data statistic per QoS flow

Parameters

bearerId	• Bearer ID
tx_pkt	• number of sent packets for the QoS flow ID
tx_pkt_drp	• number of dropped(TX) packets for the QoS flow ID
tx_bytes	• number of sent bytes for the QoS flow ID
tx_bytes_drp	• number of dropped(TX) bytes for the QoS flow ID

8.626.2 Field Documentation

8.626.2.1 uint32_t unpack_QosFlowStat_t::bearerId

8.626.2.2 uint64_t unpack_QosFlowStat_t::tx_bytes

8.626.2.3 uint64_t unpack_QosFlowStat_t::tx_bytes_drp

8.626.2.4 uint32_t unpack_QosFlowStat_t::tx_pkt

8.626.2.5 uint32_t unpack_QosFlowStat_t::tx_pkt_drp

8.627 unpack_rms_GetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask

8.627.1 Detailed Description

Parameters

enabled	• SMS wake functionality enabled – 0 - Disabled – 1 - Enabled
wake_mask	• SMS wake mask to search for incoming messages (only relevant when enabled)

8.627.2 Field Documentation

8.627.2.1 uint32_t unpack_rms_GetSMSWake_t::enabled

8.627.2.2 uint32_t unpack_rms_GetSMSWake_t::wake_mask

8.628 unpack_RMTransferStatistics_ind_t Struct Reference

Data Fields

- wds_DataULongTlv TxOkConutTlv
- wds_DataULongTlv RxOkConutTlv
- wds_DataULongLongTlv TxOkByteCountTlv
- wds_DataULongLongTlv RxOkByteCountTlv
- wds_DataULongTlv TxDropConutTlv
- wds_DataULongTlv RxDropConutTlv

8.628.1 Detailed Description

WDS packet RM Transfer Statistics data structure for individual session

Parameters

TxOkConutTlv	• Tx Ok Packet Tlv Value.
RxOkConutTlv	• Rx Ok Packet Tlv Value.
TxOkByteCount-Tlv	• Tx Ok Byte Count Packet Tlv Value.
RxOkByteCount-Tlv	• Rx Ok Byte Count Packet Tlv Value.
TxDropConutTlv	• Tx Drop Count Packet Tlv Value.
RxDropConutTlv	• Rx Drop Count Packet Tlv Value.

8.628.2 Field Documentation

8.628.2.1 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxDropConutTlv

8.628.2.2 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::RxOkByteCountTlv

8.628.2.3 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxOkConutTlv

8.628.2.4 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxDropConutTlv

8.628.2.5 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::TxOkByteCountTlv

8.628.2.6 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxOkConutTlv

8.629 unpack_sar_SLQSGetRfSarState_t Struct Reference

Data Fields

- uint32_t * pSarRFState

8.629.1 Detailed Description

Parameters

pSarRFState	• SAR RF State – QMI_SAR_RF_STATE_DEFAULT – QMI_SAR_RF_STATE_1 – QMI_SAR_RF_STATE_2 – QMI_SAR_RF_STATE_3 – QMI_SAR_RF_STATE_4 – QMI_SAR_RF_STATE_5 – QMI_SAR_RF_STATE_6 – QMI_SAR_RF_STATE_7 – QMI_SAR_RF_STATE_8

8.629.2 Field Documentation

8.629.2.1 uint32_t* unpack_sar_SLQSGetRfSarState_t::pSarRFState

8.630 unpack_sms_GetSMSCAddress_t Struct Reference

Data Fields

- uint8_t addressSize
- uint8_t * pSMSCAddress
- uint8_t typeSize
- uint8_t * pSMSCType

8.630.1 Detailed Description

Structure contains Gets the SMS center address parameters.

Parameters

addressSize	• The maximum number of characters (including NULL terminator) that the SMS center address array can contain.
pSMSCAddress	• The SMS center address represented as a NULL terminated string.
typeSize	• The maximum number of characters (including NULL terminator) that the SMS center address type array can contain.
pSMSCType	• The SMS center address type represented as a NULL terminated string.

8.630.2 Field Documentation

8.630.2.1 uint8_t unpack_sms_GetSMSCAddress_t::addressSize

8.630.2.2 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCAddress

8.630.2.3 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCType

8.630.2.4 uint8_t unpack_sms_GetSMSCAddress_t::typeSize

8.631 unpack_sms_SaveSMS_t Struct Reference

Data Fields

- uint32_t * pMessageIndex

8.631.1 Detailed Description

Parameters

pMessageIndex	• The message index assigned by the device

8.631.2 Field Documentation

8.631.2.1 uint32_t* unpack_sms_SaveSMS_t::pMessageIndex

8.632 unpack_sms_SendSMS_t Struct Reference

Data Fields

- uint16_t messageID
- uint32_t messageFailureCode

8.632.1 Detailed Description

Parameters

messageID	• WMS message ID
messageFailure-Code	• pointer to message failure code. If cause code is not provided, then value will be 0xFFFFFFFF

8.632.2 Field Documentation

8.632.2.1 uint32_t unpack_sms_SendSMS_t::messageFailureCode

8.632.2.2 uint16_t unpack_sms_SendSMS_t::messageID

8.633 unpack_sms_SetNewSMSCallback_ind_t Struct Reference

Data Fields

- struct newMTMessageTlv NewMMTlv
- struct transferRouteMessageTlv TRMessageTlv
- struct messageModeTlv MMTlv
- struct sMSEtwsMessageTlv ETWSTlv
- struct eTWSPLMNInfoTlv ETWSPLMNTlv
- struct sMSCAddressTlv SMSCTlv
- struct sMSOnIMSTlv IMSTlv

8.633.1 Detailed Description

Parameters

NewMMTlv	• MT message
TRMessageTlv	• Transfer Route MT Message • See transferRouteMessageTlv for more information
MMTlv	• Message mode • See messageModeTlv for more information
ETWSTlv	• ETWS Message • See sMSEtwsMessageTlv for more information
ETWSPLMNTlv	• ETWS PLMN Information • See eTWSPLMNInfoTlv for more information
SMSCTlv	• SMSC Address • See sMSCAddressTlv for more information
IMSTlv	• SMS on IMS • See sMSOnIMSTlv for more information

8.633.2 Field Documentation

8.633.2.1 struct eTWSPLMNInfoTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSPLMNTlv

8.633.2.2 struct sMSEtwsMessageTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSTlv

8.633.2.3 struct sMSOnIMSTlv unpack_sms_SetNewSMSCallback_ind_t::IMSTlv

8.633.2.4 struct messageModeTlv unpack_sms_SetNewSMSCallback_ind_t::MMTlv

8.633.2.5 struct newMTMessageTlv unpack_sms_SetNewSMSCallback_ind_t::NewMMTlv

8.633.2.6 struct sMSCAddressTlv unpack_sms_SetNewSMSCallback_ind_t::SMSCTlv

8.633.2.7 struct transferRouteMessageTlv unpack_sms_SetNewSMSCallback_ind_t::TRMessageTlv

8.634 unpack_sms_SetNewSMSCallback_t Struct Reference

8.635 unpack_sms_SLQSDeleteSMS_t Struct Reference

8.636 unpack_sms_SLQSGetIndicationRegister_t Struct Reference

Data Fields

- sms_getIndicationReg * pGetIndicationRegInfo

8.636.1 Detailed Description

Structure containing Parameters that provides registration state of different WMS indications.

Parameters

pGetIndication-RegInfo	• Pointer to structure of getIndicationReg – See sms_getIndicationReg for more information

8.636.2 Field Documentation

8.636.2.1 sms_getIndicationReg* unpack_sms_SLQSGetIndicationRegister_t::pGetIndicationRegInfo

8.637 unpack_sms_SLQSGetMessageWaiting_t Struct Reference

Data Fields

- sms_getMsgWaitingInfo * pGetMsgWaitingInfoResp

8.637.1 Detailed Description

Structure contain Parameter that provide information about the message waiting information.

Parameters

pGetMsg-WaitingInfoResp	• Pointer to structure of getMsgWaitingInfoResp – See sms_getMsgWaitingInfo for more information

8.637.2 Field Documentation

8.637.2.1 sms_getMsgWaitingInfo* unpack_sms_SLQSGetMessageWaiting_t::pGetMsgWaitingInfoResp

8.638 unpack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t messageTag
- uint32_t messageFormat
- uint32_t messageSize
- uint8_t message [2048]

8.638.1 Detailed Description

Parameters

messageTag	• Message tag – 0 - Read – 1 - Not read – 2 - Mobile originated and sent – 3 - Mobile originated but not yet sent
messageFormat	• Message format – 0 - CDMA (IS-637B) – 1 - 5 (Reserved) – 6 - GSM/WCDMA PP
messageSize	• Upon input the maximum number of bytes that can be written to the message array.

- Upon successful output the actual number of bytes written to the message array.

Parameters

message	• The message contents array

8.638.2 Field Documentation

8.638.2.1 uint8_t unpack_sms_SLQSGetSMS_t::message[2048]

8.638.2.2 uint32_t unpack_sms_SLQSGetSMS_t::messageFormat

8.638.2.3 uint32_t unpack_sms_SLQSGetSMS_t::messageSize

8.638.2.4 uint32_t unpack_sms_SLQSGetSMS_t::messageTag

8.639 unpack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- sms_qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig

- sms_qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig

8.639.1 Detailed Description

Structure contain parameters that Provides Information about the SMS BroadcastConfiguration

Parameters

pBroadcast-Config	• The data for 3GPP Broadcast Information(Optional).
pCDMA-BroadcastConfig	• The data for 3GPP2 Broadcast Information(Optional).

8.639.2 Field Documentation

8.639.2.1 sms_qaQmi3GPPBroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pBroadcastConfig

8.639.2.2 sms_qaQmi3GPP2BroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pCDMABroadcastConfig

8.640 unpack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t messageListSize
- qmiSmsMessageList messageList [255]

8.640.1 Detailed Description

Parameters

messageListSize	• Upon input the maximum number of elements that the message list array can contain. • Upon successful output the actual number of elements in the message list array.
messageList	• Message List • See qmiSmsMessageList for more information

8.640.2 Field Documentation

8.640.2.1 qmiSmsMessageList unpack_sms_SLQSGetSMSList_t::messageList[255]

8.640.2.2 uint32_t unpack_sms_SLQSGetSMSList_t::messageListSize

8.641 unpack_sms_SLQSGetTransLayerInfo_t Struct Reference

Data Fields

- sms_getTransLayerInfo * pGetTransLayerInfo

8.641.1 Detailed Description

Structure contain parameters that gives information about the transport layer.

Parameters

pGetTransLayer-Info	• Pointer to structure of getTransLayerInfo – See sms_getTransLayerInfo for more information

8.641.2 Field Documentation

8.641.2.1 sms_getTransLayerInfo* unpack_sms_SLQSGetTransLayerInfo_t::pGetTransLayerInfo

8.642 unpack_sms_SLQSGetTransNWRegInfo_t Struct Reference

Data Fields

- sms_getTransNWRegInfo * pGetTransNWRegInfo

8.642.1 Detailed Description

Structure containing parameters that provides transport layer network registration info.

Parameters

pGetTransNW-RegInfo	• Pointer to structure of getTransNWRegInfo – See sms_getTransNWRegInfo for more information

8.642.2 Field Documentation

8.642.2.1 sms_getTransNWRegInfo* unpack_sms_SLQSGetTransNWRegInfo_t::pGetTransNWRegInfo

8.643 unpack_sms_SLQSModifySMSStatus_t Struct Reference

8.644 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeResp * pMaxStorageSizeResp

8.644.1 Detailed Description

Structure contain parameters that provides the maximum number of messages that can be stored in the specified memory storage. Also it provides the number of slots currently available

Parameters

pMaxStorage-SizeResp	• Response parameters for SmsSLQSGetMaxStorageSize – See sms_maxStorageSizeResp for more information

8.644.2 Field Documentation

8.644.2.1 sms_maxStorageSizeResp∗ unpack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeResp

8.645 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference

Data Fields

- sms_msgProtocolResp ∗ pMessageProtocol

8.645.1 Detailed Description

Structure contain Parameters that get the message protocol currently in use for the WMS client.

Parameters

pMessage-Protocol	• Pointer to smsMsgprotocolResp – See sms_msgProtocolResp for more information

8.645.2 Field Documentation

8.645.2.1 sms_msgProtocolResp∗ unpack_sms_SLQSSmsGetMessageProtocol_t::pMessageProtocol

8.646 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference

Data Fields

- uint32_t ∗ pSmsStorage

8.646.1 Detailed Description

Structure contain Parameters that return current SMS configuration that is applied to all incoming and outgoing messages.

Parameters

pSmsStorage	• Values: – 0x01 - device's permanent memory – 0x02 - UICC

8.646.2 Field Documentation

8.646.2.1 uint32_t* unpack_sms_SLQSSwiGetSMSStorage_t::pSmsStorage

8.647 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference

Data Fields

- uint8_t storageType
- uint8_t messageMode

8.647.1 Detailed Description

Parameters

storageType	• SMS message storage type – 0 - UIM - Invalid in case of CDMA device that does not require SIM – 1 - NV
messageMode	• 0x00 - CDMA, LTE (if network type is CDMA) • 0x01 - GW, LTE (if network type is UMTS)

8.647.2 Field Documentation

8.647.2.1 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::messageMode

8.647.2.2 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::storageType

8.648 unpack_swiavms_SLQSAVMSGetSettings_t Struct Reference

Data Fields

- uint32_t OMADMEnabled
- uint8_t AutoConnect
- uint8_t AutoReboot
- uint8_t FwAutodownload
- uint8_t FwAutoUpdate
- uint8_t FwAutoSDM
- uint32_t * pPollingTimer
- PackSwiAVMSSettingsConnectionRetryTimers * pConnectionRetryTimers
- PackSwiAVMSSettingsAPNInfo * pAPNInfo
- uint8_t * pNotificationStore
- PackSwiAVMSSettingsPeriodsInfo * pPeroidsInfo
- uint32_t resultcode

8.648.1 Detailed Description

Structure that contains the session type response for AVMS Set Settings command

Parameters

resultcode	- Tlv Result Code.

8.648.2 Field Documentation

8.648.2.1 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::AutoConnect

8.648.2.2 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::AutoReboot

8.648.2.3 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwAutodownload

8.648.2.4 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwAutoSDM

8.648.2.5 uint8_t unpack_swiavms_SLQSAVMSGetSettings_t::FwAutoUpdate

8.648.2.6 uint32_t unpack_swiavms_SLQSAVMSGetSettings_t::OMADMEnabled

8.648.2.7 PackSwiAVMSSettingsAPNInfo* unpack_swiavms_SLQSAVMSGetSettings_t::pAPNInfo

8.648.2.8 PackSwiAVMSSettingsConnectionRetryTimers* unpack_swiavms_SLQSAVMSGetSettings_t::pConnectionRetryTimers

8.648.2.9 uint8_t* unpack_swiavms_SLQSAVMSGetSettings_t::pNotificationStore

8.648.2.10 PackSwiAVMSSettingsPeriodsInfo* unpack_swiavms_SLQSAVMSGetSettings_t::pPeroidsInfo

8.648.2.11 uint32_t* unpack_swiavms_SLQSAVMSGetSettings_t::pPollingTimer

8.648.2.12 uint32_t unpack_swiavms_SLQSAVMSGetSettings_t::resultcode

8.649 unpack_swiavms_SLQSAVMSSendSelection_t Struct Reference

Data Fields

- uint32_t resultcode

8.649.1 Detailed Description

Structure that contains the session type response for AVMS Selection command

Parameters

resultcode	- Tlv Result Code.

8.649.2 Field Documentation

8.649.2.1 uint32_t unpack_swiavms_SLQSAVMSSendSelection_t::resultcode

8.650 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference

Data Fields

- uint32_t resultcode

8.650.1 Detailed Description

Structure that contains the session type response for AVMS set event report command

Parameters

resultcode	- Tlv Result Code.

8.650.2 Field Documentation

8.650.2.1 uint32_t unpack_swiavms_SLQSAvmsSetEventReport_t::resultcode

8.651 unpack_swiavms_SLQSAVMSSetSettings_t Struct Reference

Data Fields

- uint32_t resultcode

8.651.1 Detailed Description

Structure that contains the session type response for AVMS Set Settings command

Parameters

resultcode	- Tlv Result Code.

8.651.2 Field Documentation

8.651.2.1 uint32_t unpack_swiavms_SLQSAVMSSetSettings_t::resultcode

8.652 unpack_swiavms_SLQSAVMSStartSession_t Struct Reference

Data Fields

- uint32_t sessionResponse
- uint32_t resultcode

8.652.1 Detailed Description

Structure that contains the session type response for AVMS start session command

Parameters

Session-Response	- OMA-DM availability CHECK for the requested item. • 0x00000001 - Available. • 0x00000002 - Not Available. • 0x00000003 - Check Timed Out.
resultcode	- Tlv Result Code.

8.652.2 Field Documentation

8.652.2.1 uint32_t unpack_swiavms_SLQSAVMSStartSession_t::resultcode

8.652.2.2 uint32_t unpack_swiavms_SLQSAVMSStartSession_t::sessionResponse

8.653 unpack_swiavms_SLQSAVMSStopSession_t Struct Reference

Data Fields

- uint32_t resultcode

8.653.1 Detailed Description

Structure that contains the session type response for AVMS Stop session command

Parameters

resultcode	- Tlv Result Code.

8.653.2 Field Documentation

8.653.2.1 uint32_t unpack_swiavms_SLQSAVMSStopSession_t::resultcode

8.654 unpack_swiloc_SwiLocGetAutoStart_t Struct Reference

Data Fields

- uint8_t function
- int function_reported
- uint8_t fix_type
- int fix_type_reported
- uint8_t max_time
- int max_time_reported
- uint32_t max_dist
- int max_dist_reported
- uint32_t fix_rate
- int fix_rate_reported

8.654.1 Detailed Description

This structure contains SWI LOC Get Auto Start setting

Parameters

function	• Setting to indicate when modem should start an automatic GNSS fix – 0 - disabled – 1 - At bootup – 2 - When NMEA port is opened
function_reported	• 0 - not reported by modem • I - reported by modem

fix_type	• Type of GNSS fix: – 1 - Default Engine mode – 2 - MS-Based – 3 - MS-Assisted – 4 - Standalone
fix_type_reported	• 0 - not reported by modem • l - reported by modem
max_time	• Maximum time allowed for the receiver to get a fix in seconds • Valid range: 1-255
max_time_reported	• 0 - not reported by modem • l - reported by modem
max_dist	• Maximum uncertainty of a fix measured by distance in meters • Valid range: 1 - 4294967280
max_dist_reported	• 0 - not reported by modem • l - reported by modem
fix_rate	• Time between fixes in seconds • Valid range: 1–65535
fix_rate_reported	• 0 - not reported by modem • l - reported by modem

8.654.2 Field Documentation

8.654.2.1 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::fix_rate

8.654.2.2 int unpack_swiloc_SwiLocGetAutoStart_t::fix_rate_reported

8.654.2.3 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::fix_type

8.654.2.4 int unpack_swiloc_SwiLocGetAutoStart_t::fix_type_reported

8.654.2.5 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::function

8.654.2.6 int unpack_swiloc_SwiLocGetAutoStart_t::function_reported

8.654.2.7 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::max_dist

8.654.2.8 int unpack_swiloc_SwiLocGetAutoStart_t::max_dist_reported

8.654.2.9 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::max_time

8.654.2.10 int unpack_swiloc_SwiLocGetAutoStart_t::max_time_reported

8.655 unpack_swioma_SLQSOMADMAlertCallback_ind_t Struct Reference

Data Fields

- uint32_t eventType
- unpack_omaDmFotaTlv_t SessionInfoFota
- unpack_omaDmConfigTlv_t SessionInfoConfig
- unpack_omaDmNotificationsTlv_t SessionInfoNotification

8.655.1 Detailed Description

Structure that contains OMA indication information based on eventType Strucutres for which the event is not valid will have values set to 0

Parameters

eventType	• 0x00 - SWIOMA-DM FOTA • 0x01 - SWIOMA-DM Config • 0x02 - SWIOMA-DM Notification
SessionInfo-Fota[OUT]	• See unpack_omaDmFotaTlv_t for more information
SessionInfo-Config[OUT]	• See unpack_omaDmConfigTlv_t for more information
SessionInfo-Notification[OUT]	• See unpack_omaDmNotificationsTlv_t for more information

8.655.2 Field Documentation

8.655.2.1 uint32_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::eventType

8.655.2.2 unpack_omaDmConfigTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoConfig

8.655.2.3 unpack_omaDmFotaTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoFota

8.655.2.4 unpack_omaDmNotificationsTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoNotification

8.656 unpack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference

Data Fields

- uint8_t Status
- uint16_t UpdateCompleteStatus

- uint8_t Severity
- uint16_t SourceLength
- uint8_t Source [255]
- uint16_t PkgNameLength
- uint8_t PkgName [255]
- uint16_t PkgDescLength
- uint8_t PkgDescription [255]
- uint16_t DateLength
- uint8_t Date [255]
- uint16_t TimeLength
- uint8_t Time [255]
- uint8_t SessionType
- uint8_t SessionState
- uint16_t RetryCount

8.656.1 Detailed Description

Structure that contains the session type for OMA get session info unpack command Also used as input parameter to specify the size of variable parameters. (ref. notes)

Parameters

Status	• 1 Byte parameter indicating status – 0x01 - No Firmware available – 0x02 - Query Firmware Download – 0x03 - Firmware Downloading – 0x04 - Firmware Downloaded – 0x05 - Query Firmware Update – 0x06 - Firmware Updating – 0x07 - Firmware Updated
Update-CompleteStatus	• 2 byte parameter indicating Update Complete Status – See qaGobiApiTableSwiOMADMUpdateCompleteStatus.h Update Complete Status
Severity	• 1 byte parameter indicating severity – 0x01 - Mandatory – 0x02 - Optional
SourceLength	• 2 byte parameter indicating Length of Vendor Name String in Bytes.
Source	• Variable length parameter indicating Vendor Name in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information
PkgNameLength	• 2 byte parameter indicating Length of Package Name String in Bytes.
PkgName	• Variable length parameter indicating Package Name in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information

PkgDescLength	• 2 byte parameter indicating Length of Package Description String in Bytes.
PkgDescription	• Variable length parameter indicating Package Description in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information
DateLength	• 2 byte parameter indicating Length of Package Description String in Bytes.
Date	• Variable length parameter indicating Package Description in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information
TimeLength	• 2 byte parameter indicating Length of Time String in Bytes.
Time	• Variable length parameter indicating Time String in ASCII • See LITEQMI_MAX_SWIOMA_STR_LEN for more information
SessionType	• 1 byte parameter reflects the last session started for Sprint – 0x00 - No session since boot – 0x01 - Sprint CI-DC Session – 0x02 - Sprint CI-PRL Session – 0x03 - Sprint CI-FUMO Session – 0x04 - Sprint HFA-DC Session – 0x05 - Sprint HFA-PRL Session – 0x06 - Sprint HFA-FUMO Session – 0x07 - Sprint NI Session
SessionState	• 1 byte parameter indicating session state – 0x01 - idle – 0x02 - active – 0x03 - pending
RetryCount	• 1 byte parameter indicating retries left count – valid values 0 to 6

8.656.2 Field Documentation

8.656.2.1 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Date[255]

8.656.2.2 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::DateLength

8.656.2.3 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescLength

8.656.2.4 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescription[255]

8.656.2.5 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgName[255]

8.656.2.6 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgNameLength

8.656.2.7 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::RetryCount

8.656.2.8 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionState

8.656.2.9 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.656.2.10 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Severity

8.656.2.11 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Source[255]

8.656.2.12 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SourceLength

8.656.2.13 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Status

8.656.2.14 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Time[255]

8.656.2.15 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::TimeLength

8.656.2.16 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::UpdateCompleteStatus

8.657 unpack_swioma_SLQSOMADMGetSettings_t Struct Reference

Data Fields

- uint32_t OMADMEnabled
- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t Autosdm
- uint8_t FwAutoCheck

8.657.1 Detailed Description

Structure containing the OMA DM settings retrieved from the device

Parameters

out	*OMADMEnabled*	• Optional 4 byte parameter indicating OMADM service enabled – 0x00000001 - Client-initiated device configuration – 0x00000002 - Network-initiated device configuration – 0x00000010 - Client-initiated FUMO – 0x00000020 - Network-initiated FUMO • function SLQSOMADMGetSettings2() returns a default value 0xFFFFFFFF in case this parameter is not returned by the modem.

out	*FOTAdownload*	• Optional 1 Byte parameter indicating support for FOTA Automatic download – 0x00 - Host permission required before downloading – 0x01 - Automatically start downloading, no host permission required – 0x02 - Automatically start downloading, while not roaming – 0x03 - Automatically reject download – 0x04 - Automatically reject download with “Enterprise Reject Policy”

- function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.

Parameters

out	*FOTAUpdate*	• Optional 1 byte parameter indicating FOTA Automatic update – 0x00 - User permission required before updating firmware – 0x01 - No user permission required before updating firmware – 0x02 - User permission required, auto update on power up • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
out	*Autosdm*	• Optional 1 byte parameter indicating OMA Automatic UI Alert Response – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.
out	*FwAutoCheck*	• Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response – 0x00 - Disabled – 0x01 - Enabled • function SLQSOMADMGetSettings2() returns a default value 0xFF in case this parameter is not returned by the modem.

8.657.2 Field Documentation

8.657.2.1 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::Autosdm

8.657.2.2 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAdownload

8.657.2.3 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAUpdate

8.657.2.4 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FwAutoCheck

8.657.2.5 uint32_t unpack_swioma_SLQSOMADMGetSettings_t::OMADMEnabled

8.658 unpack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t FwAvailability

8.658.1 Detailed Description

Structure that contains the responses for OMA start session command

Parameters

out	*FwAvailability*	• OMA-DM CHECK FW Available – 0x00000001 - FW Available. For CIDC and CIPRL, this value will be returned by the modem. CIDC and CIPRL are asynchronous OMADM sessions. – 0x00000002 - FW Not Available – 0x00000003 - FW Check Timed Out

8.658.2 Field Documentation

8.658.2.1 uint32_t unpack_swioma_SLQSOMADMStartSession_t::FwAvailability

8.659 unpack_uim_ChangePin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t Tlvresult

8.659.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information.
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • 0xFFFFFFFF, if unavailable

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.659.2 Field Documentation

8.659.2.1 uim_encryptedPIN1∗ unpack_uim_ChangePin_t::pEncryptedPIN1

8.659.2.2 uint32_t∗ unpack_uim_ChangePin_t::pIndicationToken

8.659.2.3 uim_remainingRetries∗ unpack_uim_ChangePin_t::pRemainingRetries

8.659.2.4 uint16_t unpack_uim_ChangePin_t::Tlvresult

8.660 unpack_uim_GetCardStatus_t Struct Reference

Data Fields

- uim_cardStatus ∗ pCardStatus
- uim_hotSwapStatus ∗ pHotSwapStatus
- uint16_t Tlvresult

8.660.1 Detailed Description

This structure contains information of the response parameters associated with a Get Card Status API.

Parameters

pCard-Status(optional)	• See uim_cardStatus for more information.
pHotSwap-Status(optional)	• See uim_hotSwapStatus for more information.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.660.2 Field Documentation

8.660.2.1 uim_cardStatus∗ unpack_uim_GetCardStatus_t::pCardStatus

8.660.2.2 uim_hotSwapStatus∗ unpack_uim_GetCardStatus_t::pHotSwapStatus

8.660.2.3 uint16_t unpack_uim_GetCardStatus_t::Tlvresult

8.661 unpack_uim_ReadTransparent_t Struct Reference

Data Fields

- uim_cardResult ∗ pCardResult
- uim_readResult ∗ pReadResult
- uint32_t ∗ pIndicationToken

- uint8_t * pEncryptedData
- uint16_t Tlvresult

8.661.1 Detailed Description

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

pCardResult	• See uim_cardResult for more information.
pReadResult	• See uim_readResult for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.
pEncrypted-Data(optional)	• Encrypted Data. • Indicates whether the data from the card passed in read_result is encrypted.

Note

Using NULL for the pointers would make sure that the parameter is not added to the request.

8.661.2 Field Documentation

8.661.2.1 uim_cardResult* unpack_uim_ReadTransparent_t::pCardResult

8.661.2.2 uint8_t* unpack_uim_ReadTransparent_t::pEncryptedData

8.661.2.3 uint32_t* unpack_uim_ReadTransparent_t::pIndicationToken

8.661.2.4 uim_readResult* unpack_uim_ReadTransparent_t::pReadResult

8.661.2.5 uint16_t unpack_uim_ReadTransparent_t::Tlvresult

8.662 unpack_uim_SetPinProtection_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t Tlvresult

8.662.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information.
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • 0xFFFFFFFF, if unavailable

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.662.2 Field Documentation

8.662.2.1 uim_encryptedPIN1∗ unpack_uim_SetPinProtection_t::pEncryptedPIN1

8.662.2.2 uint32_t∗ unpack_uim_SetPinProtection_t::pIndicationToken

8.662.2.3 uim_remainingRetries∗ unpack_uim_SetPinProtection_t::pRemainingRetries

8.662.2.4 uint16_t unpack_uim_SetPinProtection_t::TlvResult

8.663 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference

Data Fields

- slots_t slotsstatusChange
- uint8_t bNumberOfPhySlots

8.663.1 Detailed Description

Structure consist of card status params

Parameters

slotsstatus-Change	• See slot_t for more information
bNumberOfPhy-Slots	• Number of Physical Slot(s)

8.663.2 Field Documentation

8.663.2.1 uint8_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::bNumberOfPhySlots

8.663.2.2 slots_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::slotsstatusChange

8.664 unpack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_authenticateResult * pAuthenticateResult
- uint32_t * pIndicationToken

8.664.1 Detailed Description

This structure contains information of the response parameters associated with a SLQSUIMAuthenticate.

Parameters

pCard-Result(optional)	• See uim_cardResult for more information.
pAuthenticate-Result(optional)	• See uim_authenticateResult for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.664.2 Field Documentation

8.664.2.1 uim_authenticateResult* unpack_uim_SLQSUIMAuthenticate_t::pAuthenticateResult

8.664.2.2 uim_cardResult* unpack_uim_SLQSUIMAuthenticate_t::pCardResult

8.664.2.3 uint32_t* unpack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.665 unpack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries

8.665.1 Detailed Description

This structure contains information of the response parameters associated with a Depersonalization.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.665.2 Field Documentation

8.665.2.1 uim_remainingRetries* unpack_uim_SLQSUIMDepersonalization_t::pRemainingRetries

8.666 unpack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask

8.666.1 Detailed Description

Parameters

eventMask	- bit 0 - card status • bit 1 - SAP connection • bit 4 - physical slot status

8.666.2 Field Documentation

8.666.2.1 uint32_t unpack_uim_SLQSUIMEventRegister_t::eventMask

8.667 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint8_t * pAutoSelection
- uim_personalizationStatus * pPersonalizationStatus
- uint8_t * pHaltSubscription

8.667.1 Detailed Description

This structure contains information of the response parameters associated with Get Configuration.

Parameters

pAuto-Selection(optional)	• Indicates whether the modem is configured to automatically select the provisioning sessions at powerup. • Valid values – 0 - Automatic provisioning is off – 1 - Automatic provisioning is on
p-Personalization-Status(optional)	• See uim_personalizationStatus for more information.

pHalt-Subscription(optional)	• Indicates if the modem is configured to publish the subscription after successful initialization. • Valid values – 0 - Modem proceeds with publishing the subscription – 1 - Modem does not publish the subscription

8.667.2 Field Documentation

8.667.2.1 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pAutoSelection

8.667.2.2 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pHaltSubscription

8.667.2.3 uim_personalizationStatus* unpack_uim_SLQSUIMGetConfiguration_t::pPersonalizationStatus

8.668 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_fileAttributes * pFileAttributes
- uint32_t * pIndicationToken

8.668.1 Detailed Description

This structure contains information of the response parameters associated with a Get File Attributes

Parameters

pCard-Result(optional)	• See uim_cardResult for more information.
pFile-Attributes(optional)	• See uim_fileAttributes for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result must be provided in a subsequent indication.

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.668.2 Field Documentation

8.668.2.1 uim_cardResult* unpack_uim_SLQSUIMGetFileAttributes_t::pCardResult

8.668.2.2 uim_fileAttributes* unpack_uim_SLQSUIMGetFileAttributes_t::pFileAttributes

8.668.2.3 uint32_t* unpack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.669 unpack_uim_SLQSUIMGetSlotsStatus_t Struct Reference

Data Fields

- uint8_t * pNumberOfPhySlot
- slots_t * pUimSlotsStatus

8.669.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

pNumberOfPhySlot	• Number of sets of the Slot Status.
pUimSlotsStatus	• Slots Status See slots_t for more information..

8.669.2 Field Documentation

8.669.2.1 uint8_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pNumberOfPhySlot

8.669.2.2 slots_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pUimSlotsStatus

8.670 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_refreshevent * pRefreshEvent

8.670.1 Detailed Description

This structure contains information of the response parameters associated with a Refresh Get Last Event.

Parameters

refreshEvent(-Optional)	• See uim_refreshevent for more information.

8.670.2 Field Documentation

8.670.2.1 uim_refreshevent* unpack_uim_SLQSUIMRefreshGetLastEvent_t::pRefreshEvent

8.671 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference

Data Fields

- uim_cardStatus * pCardStatus

8.671.1 Detailed Description

This structure contains information about Status change callback.

Parameters

pCardStatus	Card Status • See uim_cardStatus for more information.

8.671.2 Field Documentation

8.671.2.1 uim_cardStatus* unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t::pCardStatus

8.672 unpack_uim_UnblockPin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t TlvresultTlvresult

8.672.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information.
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • 0xFFFFFFFF, if unavailable

8.672.2 Field Documentation

8.672.2.1 uim_encryptedPIN1* unpack_uim_UnblockPin_t::pEncryptedPIN1

8.672.2.2 uint32_t* unpack_uim_UnblockPin_t::pIndicationToken

8.672.2.3 uim_remainingRetries* unpack_uim_UnblockPin_t::pRemainingRetries

8.672.2.4 uint16_t unpack_uim_UnblockPin_t::Tlvresult

8.673 unpack_uim_VerifyPin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t TlvResult

8.673.1 Detailed Description

This structure contains information of the response parameters associated with a set of PIN related API's.

Parameters

pRemaining-Retries(optional)	• See uim_remainingRetries for more information.
pEncryptedPI-N1(optional)	• See uim_encryptedPIN1 for more information.
pIndication-Token(optional)	• Response in Indication. • When this TLV is present, it indicates that the result is provided in a subsequent indication. • 0xFFFFFFFF, if unavailable

Note

Using NULL for the pointers would make sure that the parameter is not returned.

8.673.2 Field Documentation

8.673.2.1 uim_encryptedPIN1* unpack_uim_VerifyPin_t::pEncryptedPIN1

8.673.2.2 uint32_t* unpack_uim_VerifyPin_t::pIndicationToken

8.673.2.3 uim_remainingRetries* unpack_uim_VerifyPin_t::pRemainingRetries

8.673.2.4 uint16_t unpack_uim_VerifyPin_t::TlvResult

8.674 unpack_voice_SLQSOriginateUSSD_t Struct Reference

Data Fields

- uint16_t * pfailureCause
- voice_alphaIDInfo * pAlphaIDInfo
- struct voice_USSInfo * pUSSDInfo
- uint8_t * pCcResultType
- uint8_t * pCallId
- voice_ccSUPSType * pCCSuppsType

8.674.1 Field Documentation

8.674.1.1 voice_alphaIDInfo* unpack_voice_SLQSOriginateUSSD_t::pAlphaIDInfo

8.674.1.2 uint8_t* unpack_voice_SLQSOriginateUSSD_t::pCallId

8.674.1.3 uint8_t* unpack_voice_SLQSOriginateUSSD_t::pCcResultType

8.674.1.4 voice_ccSUPSType* unpack_voice_SLQSOriginateUSSD_t::pCCSuppsType

8.674.1.5 uint16_t* unpack_voice_SLQSOriginateUSSD_t::pfailureCause

8.674.1.6 struct voice_USSInfo* unpack_voice_SLQSOriginateUSSD_t::pUSSDInfo

8.675 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.675.1 Detailed Description

Contains the parameters passed for pack voice Answer Call.

Parameters

pCallId	• Unique call identifier for the call that must be answered.

8.675.2 Field Documentation

8.675.2.1 uint8_t* unpack_voice_SLQSVoiceAnswerCall_t::pCallId

8.676 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID

8.676.1 Detailed Description

This structure contains Voice Burst DTMF Information for unpack.

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. A burst DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device

8.676.2 Field Documentation

8.676.2.1 uint8_t* unpack_voice_SLQSVoiceBurstDTMF_t::pCallID

8.677 unpack_voice_SLQSVoiceDialCall_t Struct Reference

Data Fields

- uint8_t * pCallID
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResultType
- voice_ccSUPSType * pCCSUPSType

8.677.1 Detailed Description

This structure contains Voice Call Response Parameters

Parameters

pCallID(optional)	• Unique call identifier for the dialed call
pAlphaID-Info(optional)	• Pointer to structure of alphaIDInfo – See voice_alphaIDInfo for more information
pCCResult-Type(optional)	• Call Control Result Type. – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service
pCCSUPS-Type(optional)	• Pointer to structure of ccSUPSType • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

8.677.2 Field Documentation

8.677.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceDialCall_t::pAlphaIDInfo

8.677.2.2 uint8_t* unpack_voice_SLQSVoiceDialCall_t::pCallID

8.677.2.3 uint8_t* unpack_voice_SLQSVoiceDialCall_t::pCCResultType

8.677.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceDialCall_t::pCCSUPSType

8.678 unpack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.678.1 Detailed Description

Parameters

pCallId	• Unique call identifier for the call that must be ended

8.678.2 Field Documentation

8.678.2.1 uint8_t* unpack_voice_SLQSVoiceEndCall_t::pCallId

8.679 unpack_voice_SLQSVoiceGetAllCallInfo_t Struct Reference

Data Fields

- voice_arrCallInfo * pArrCallInfo
- voice_arrRemotePartyNum * pArrRemotePartyNum
- voice_arrRemotePartyName * pArrRemotePartyName
- voice_arrAlertingType * pArrAlertingType
- voice_arrUUSInfo * pArrUUSInfo
- voice_arrSvcOption * pArrSvcOption
- uint8_t * pOTASPStatus
- uint8_t * pVoicePrivacy
- voice_arrCallEndReason * pArrCallEndReason
- voice_arrAlphaID * pArrAlphaID
- voice_arrConnectPartyNum * pArrConnectPartyNum
- voice_arrDiagInfo * pArrDiagInfo
- voice_arrCalledPartyNum * pArrCalledPartyNum
- voice_arrRedirPartyNum * pArrRedirPartyNum
- voice_arrAlertingPattern * pArrAlertingPattern

8.679.1 Detailed Description

This structure contains information about the response parameters with all the calls originating or terminating from a particular device.

Parameters

pArrCallInfo(optional)	• See voice_arrCallInfo for more information.
pArrRemotePartyNum(optional)	• See voice_arrRemotePartyNum for more information.
pArrRemotePartyName(optional)	• See voice_arrRemotePartyName for more information.
pArrAlertingType(optional)	• See voice_arrAlertingType for more information.
pArrUUSInfo(optional)	• See voice_arrUUSInfo for more information.

pArrSvcOption(optional)	• See voice_arrSvcOption for more information.
pOTASPStatus(optional)	• OTASP status for the OTASP call. • Applicable only for 3GPP2 devices. – 0x00 - OTASP_STATUS_SPL_UNLOCKED - SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED - SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED - A-key exchanged; only for user-initiated OTASP – 0x03 - OTASP_STATUS_SSD_UPDATED - SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP (OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP (OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP (OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP (OTAPA) – 0xFF - Not Available
pVoicePrivacy(optional)	• Voice Privacy. • Values. – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy – 0xFF - Not Available
pArrCallEndReason(optional)	• See voice_arrCallEndReason for more information.
pArrAlphaID(optional)	• See voice_arrAlphaID for more information.
pArrConnectPartyNum(optional)	• See voice_arrConnectPartyNum for more information.
pArrDiagInfo(optional)	• See voice_arrDiagInfo for more information.

pArrCalledPartyNum(optional)	• See voice_arrCalledPartyNum for more information.
pArrRedirPartyNum(optional)	• See voice_arrRedirPartyNum for more information.
pArrAlertingPattern(optional)	• See voice_arrAlertingPattern for more information.

8.679.2 Field Documentation

8.679.2.1 voice_arrAlertingPattern∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingPattern

8.679.2.2 voice_arrAlertingType∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingType

8.679.2.3 voice_arrAlphaID∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlphaID

8.679.2.4 voice_arrCalledPartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCalledPartyNum

8.679.2.5 voice_arrCallEndReason∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallEndReason

8.679.2.6 voice_arrCallInfo∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallInfo

8.679.2.7 voice_arrConnectPartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrConnectPartyNum

8.679.2.8 voice_arrDiagInfo∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrDiagInfo

8.679.2.9 voice_arrRedirPartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRedirPartyNum

8.679.2.10 voice_arrRemotePartyName∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyName

8.679.2.11 voice_arrRemotePartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyNum

8.679.2.12 voice_arrSvcOption∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrSvcOption

8.679.2.13 voice_arrUUSInfo∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrUUSInfo

8.679.2.14 uint8_t∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pOTASPStatus

8.679.2.15 uint8_t∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pVoicePrivacy

8.680 unpack_voice_SLQSVoiceGetCallBarring_t Struct Reference

Data Fields

- uint8_t ∗ pSvcClass
- uint16_t ∗ pFailCause
- voice_alphaIDInfo ∗ pAlphaIDInfo
- uint8_t ∗ pCCResType
- uint8_t ∗ pCallID
- voice_ccSUPSType ∗ pCCSUPSType

8.680.1 Detailed Description

This structure contains Voice Get Call Barring Response Parameters

Parameters

pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.680.2 Field Documentation

8.680.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallBarring_t::pAlphaIDInfo

8.680.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCallID

8.680.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCCResType

8.680.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallBarring_t::pCCSUPSType

8.680.2.5 uint16_t* unpack_voice_SLQSVoiceGetCallBarring_t::pFailCause

8.680.2.6 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.681 unpack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference

Data Fields

- voice_getCallFWInfo * pGetCallFWInfo
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType
- voice_getCallFWExtInfo * pGetCallFWExtInfo

8.681.1 Detailed Description

This structure contains Voice Get Call Forwarding Status Unpack Parameters

Parameters

pGetCallFWInfo	• Pointer to structure of getCallFWInfo (optional) – See voice_getCallFWInfo for more information
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information
pGetCallFWExt-Info	• Pointer to structure of getCallFWExtInfo (optional) – See voice_getCallFWExtInfo for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.681.2 Field Documentation

8.681.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pAlphaIDInfo

8.681.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCallID

8.681.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCResType

8.681.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCSUPSType

8.681.2.5 uint16_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pFailCause

8.681.2.6 voice_getCallFWExtInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWExtInfo

8.681.2.7 voice_getCallFWInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWInfo

8.682 unpack_voice_SLQSVoiceGetCallInfo_t Struct Reference

Data Fields

- voice_callInfo * pCallInfo
- voice_remotePartyNum * pRemotePartyNum
- uint16_t * pSrvOpt
- uint8_t * pVoicePrivacy
- uint8_t * pOTASPStatus
- voice_remotePartyName * pRemotePartyName
- voice_UUSInfo * pUUSInfo
- uint8_t * pAlertType
- voice_alphaIDInfo * pAlphaIDInfo
- voice_connectNumInfo * pConnectNumInfo
- voice_diagInfo * pDiagInfo
- uint32_t * pAlertingPattern

8.682.1 Detailed Description

This structure contains information of the unpack parameters associated with a call.

Parameters

pCall-Info(optional)	• See voice_callInfo for more information.
pRemoteParty-Num(optional)	• See voice_remotePartyNum for more information.
pSrvOpt	• Service option(optional) • Applicable only for 3GPP2 devices. • See Table8 qaGobiApiTableServiceOptions.h for standard service option number assignments.

pVoicePrivacy	• Voice Privacy.(optional) • Applicable only for 3GPP2 devices. • Values. – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy – 0xFF - Not Available
pOTASPStatus	• OTASP status for the OTASP call.(optional) • Applicable only for 3GPP2 devices. – 0x00 - OTASP_STATUS_SPL_UNLOCKED - SPL unlocked; only for user-initiated OTASP – 0x01 - OTASP_STATUS_SPRC_RETRIES_EXCEEDED - SPC retries exceeded; only for user-initiated OTASP – 0x02 - OTASP_STATUS_AKEY_EXCHANGED - A-key exchanged; only for user-initiated OTASP – 0x03 - OTASP_STATUS_SSD_UPDATED - SSD updated; for both user-initiated OTASP and network-initiated OTASP (OTAPA) – 0x04 - OTASP_STATUS_NAM_DOWNLOADED - NAM downloaded; only for user-initiated OTASP – 0x05 - OTASP_STATUS_MDN_DOWNLOADED - MDN downloaded; only for user-initiated OTASP – 0x06 - OTASP_STATUS_IMSI_DOWNLOADED - IMSI downloaded; only for user-initiated OTASP – 0x07 - OTASP_STATUS_PRL_DOWNLOADED - PRL downloaded; only for user-initiated OTASP – 0x08 - OTASP_STATUS_COMMITTED - Commit successful; only for user-initiated OTASP – 0x09 - OTASP_STATUS_OTAPA_STARTED - OTAPA started; only for network-initiated OTASP (OTAPA) – 0x0A - OTASP_STATUS_OTAPA_STOPPED - OTAPA stopped; only for network-initiated OTASP (OTAPA) – 0x0B - OTASP_STATUS_OTAPA_ABORTED - OTAPA aborted; only for network-initiated OTASP (OTAPA) – 0x0C - OTASP_STATUS_OTAPA_COMMITTED - OTAPA committed; only for network-initiated OTASP (OTAPA) – 0xFF - Not Available
pRemoteParty-Name(optional)	• Applicable only for 3GPP devices. • See voice_remotePartyName for more information.
pUUS-Info(optional)	• Applicable only for 3GPP devices. • See voice_UUSInfo for more information.
pAlert-Type(optional)	• Alerting type. • Applicable only for 3GPP devices. – 0x00 - ALERTING_LOCAL - Local – 0x01 - ALERTING_REMOTE - Remote – 0xFF - Not Available

pAlphaID-Info(optional)	• Applicable only for 3GPP devices. • See voice_alphaIDInfo for more information.
pConnectNum-Info(optional)	• See voice_connectNumInfo for more information.
pDiag-Info(optional)	• See voice_diagInfo for more information.
pAlertingPattern	• Alerting pattern.(optional) – 0x00 - QMI_VOICE_ALERTING_PATTERN_1 - Pattern 1 – 0x01 - QMI_VOICE_ALERTING_PATTERN_2 - Pattern 2 – 0x02 - QMI_VOICE_ALERTING_PATTERN_3 - Pattern 3 – 0x04 - QMI_VOICE_ALERTING_PATTERN_5 - Pattern 5 – 0x05 - QMI_VOICE_ALERTING_PATTERN_6 - Pattern 6 – 0x06 - QMI_VOICE_ALERTING_PATTERN_7 - Pattern 7 – 0x07 - QMI_VOICE_ALERTING_PATTERN_8 - Pattern 8 – 0x08 - QMI_VOICE_ALERTING_PATTERN_9 - Pattern 9 – 0xFF - Not Available

8.682.2 Field Documentation

8.682.2.1 uint32_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertingPattern

8.682.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertType

8.682.2.3 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pAlphaIDInfo

8.682.2.4 voice_callInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pCallInfo

8.682.2.5 voice_connectNumInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pConnectNumInfo

8.682.2.6 voice_diagInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pDiagInfo

8.682.2.7 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pOTASPStatus

8.682.2.8 voice_remotePartyName* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyName

8.682.2.9 voice_remotePartyNum* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyNum

8.682.2.10 uint16_t* unpack_voice_SLQSVoiceGetCallInfo_t::pSrvOpt

8.682.2.11 voice_UUSInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pUUSInfo

8.682.2.12 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pVoicePrivacy

8.683 unpack_voice_SLQSVoiceGetCallWaiting_t Struct Reference

Data Fields

- uint8_t * pSvcClass
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.683.1 Detailed Description

This structure contains Voice Get Call Waiting Unpack Parameters

Parameters

pSvcClass	• Service class is a combination (sum) of information class constants (optional) • See qaGobiApiTableSupServiceInfoClasses.h for service classes. • Service Class is set to 0 if call waiting is not active for any of the information classes. • 0xFF,if Not Available
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.683.2 Field Documentation

8.683.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallWaiting_t::pAlphaIDInfo

8.683.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCallID

8.683.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCResType

8.683.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCSUPSType

8.683.2.5 uint16_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pFailCause

8.683.2.6 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.684 unpack_voice_SLQSVoiceGetCLIP_t Struct Reference

Data Fields

- voice_CLIPResp * pCLIPResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.684.1 Detailed Description

This structure contains Voice Get Calling Line Identification Presentation(CLIP) Response Parameters

Parameters

pCLIPResp	• Pointer to structure of CLIPResp (optional) – See voice_CLIPResp for more information
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.

pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.684.2 Field Documentation

8.684.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIP_t::pAlphaIDInfo

8.684.2.2 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCallID

8.684.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCCResType

8.684.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIP_t::pCCSUPSType

8.684.2.5 voice_CLIPResp* unpack_voice_SLQSVoiceGetCLIP_t::pCLIPResp

8.684.2.6 uint16_t* unpack_voice_SLQSVoiceGetCLIP_t::pFailCause

8.685 unpack_voice_SLQSVoiceGetCLIR_t Struct Reference

Data Fields

- voice_CLIRResp * pCLIRResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.685.1 Detailed Description

This structure contains Voice Get Calling Line Identification Restriction (CLIR) Response Parameters

Parameters

pCLIRResp	• Pointer to structure of CLIRResp (optional) – See voice_CLIRResp for more information
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information

pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.685.2 Field Documentation

8.685.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIR_t::pAlphaIDInfo

8.685.2.2 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCallID

8.685.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCCResType

8.685.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIR_t::pCCSUPSType

8.685.2.5 voice_CLIRResp* unpack_voice_SLQSVoiceGetCLIR_t::pCLIRResp

8.685.2.6 uint16_t* unpack_voice_SLQSVoiceGetCLIR_t::pFailCause

8.686 unpack_voice_SLQSVoiceGetCNAP_t Struct Reference

Data Fields

- voice_CNAPResp * pCNAPResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.686.1 Detailed Description

This structure contains Voice Get Calling Name Presentation(CNAP) Response Parameters

Parameters

pCNAPResp	• Pointer to structure of CNAPResp (optional) – See voice_CNAPResp for more information
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.686.2 Field Documentation

8.686.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCNAP_t::pAlphaIDInfo

8.686.2.2 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCallID

8.686.2.3 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCCResType

8.686.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCNAP_t::pCCSUPSType

8.686.2.5 voice_CNAPResp* unpack_voice_SLQSVoiceGetCNAP_t::pCNAPResp

8.686.2.6 uint16_t* unpack_voice_SLQSVoiceGetCNAP_t::pFailCause

8.687 unpack_voice_SLQSVoiceGetCOLP_t Struct Reference

Data Fields

- voice_COLPResp * pCOLPResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.687.1 Detailed Description

This structure contains Voice Get Connected Line Identification Presentation(COLP) Response Parameters

Parameters

pCOLPResp	• Pointer to structure of COLPResp (optional) – See voice_COLPResp for more information
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.687.2 Field Documentation

8.687.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLP_t::pAlphaIDInfo

8.687.2.2 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCallID

8.687.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCCResType

8.687.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLP_t::pCCSUPSType

8.687.2.5 voice_COLPResp* unpack_voice_SLQSVoiceGetCOLP_t::pCOLPResp

8.687.2.6 uint16_t* unpack_voice_SLQSVoiceGetCOLP_t::pFailCause

8.688 unpack_voice_SLQSVoiceGetCOLR_t Struct Reference

Data Fields

- voice_COLRResp * pCOLRResp
- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.688.1 Detailed Description

This structure contains Voice Get Connected Line Identification Restriction(COLR) Response Parameters

Parameters

pCOLRResp	• Pointer to structure of COLRResp (optional) – See voice_COLRResp for more information
pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.

pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.688.2 Field Documentation

8.688.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLR_t::pAlphaIDInfo

8.688.2.2 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCallID

8.688.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCCResType

8.688.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLR_t::pCCSUPSType

8.688.2.5 voice_COLRResp* unpack_voice_SLQSVoiceGetCOLR_t::pCOLRResp

8.688.2.6 uint16_t* unpack_voice_SLQSVoiceGetCOLR_t::pFailCause

8.689 unpack_voice_SLQSVoiceGetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnswerStat
- voice_airTimer * pAirTimerCnt
- voice_roamTimer * pRoamTimerCnt
- uint8_t * pCurrTTYMode
- voice_prefVoiceSO * pCurPrefVoiceSO
- voice_curAMRConfig * pCurAMRConfig
- uint8_t * pCurVoicePrivacyPref
- uint8_t * pCurVoiceDomainPref

8.689.1 Detailed Description

This structure contains Voice Get Configuration Response Parameters.

Parameters

pAutoAnswer-Stat(optional)	• Auto Answer Status • Value returned is read from NV_AUTO_ANSWER_I. – 0x00 - Disabled – 0x01 - Enabled – 0xFF - Not Available

pAirTimer-Cnt(optional)	• Air Timer Count • Value returned is read from NV_AIR_CNT_I. • See voice_airTimer for more information
pRoamTimer-Cnt(optional)	• Roam Timer Count • Value returned is read from NV_ROAM_CNT_I. • See voice_roamTimer for more information
pCurrTTY-Mode(optional)	• Current TTY Mode • Value returned is read from NV_TTY_I. – 0x00 - TTY_MODE_FULL - Full – 0x01 - TTY_MODE_VCO - Voice carry over – 0x02 - TTY_MODE_HCO - Hearing carry over – 0x03 - TTY_MODE_OFF - Off – 0xFF - Not Available
pCurPrefVoiceS-O(optional)	• Current Preferred Voice SO • Value returned is read from NV_PREF_VOICE_SO_I. • See voice_prefVoiceSO for more information
pCurAMR-Config(optional)	• Current Adaptive Multi-Rate Configuration. • Values returned are read from NV_GSM_ARM_CALL_CONFIG_I and NV_UMTS_AMR_CODEC_PREFERENCE_CONFIG_I. • See voice_curAMRConfig for more information
pCurVoice-Privacy-Pref(optional)	• Current Voice Privacy Preference • Value returned is read from NV_VOICE_PRIV_I. – 0x00 - Standard privacy – 0x01 - Enhanced privacy – 0xFF - Not Available
pCurVoice-Domain-Pref(optional)	• Current Voice Domain Preference. – 0x00 - Circuit-switched (CS) only – 0x01 - Packet-switched (PS) only – 0x02 - CS is preferred; PS is secondary – 0x03 - PS is preferred; CS is secondary – 0xFF - Not Available

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.689.2 Field Documentation

8.689.2.1 voice_airTimer* unpack_voice_SLQSVoiceGetConfig_t::pAirTimerCnt

8.689.2.2 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pAutoAnswerStat

8.689.2.3 voice_curAMRConfig* unpack_voice_SLQSVoiceGetConfig_t::pCurAMRConfig

8.689.2.4 voice_prefVoiceSO* unpack_voice_SLQSVoiceGetConfig_t::pCurPrefVoiceSO

8.689.2.5 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurrTTYMode

8.689.2.6 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoiceDomainPref

8.689.2.7 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoicePrivacyPref

8.689.2.8 voice_roamTimer* unpack_voice_SLQSVoiceGetConfig_t::pRoamTimerCnt

8.690 unpack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint16_t * pFailCause

8.690.1 Detailed Description

This structure contains Failure cause Information. Populated when API Fails.

Parameters

pFailCause	• Supplementary service failure causes (optional, supply NULL if not required). • See Table8 qaGobiApiTableVoiceCallEndReasons.h for supplementary services failure cause – 0xFFFF is the value when the information is not received from device

8.690.2 Field Documentation

8.690.2.1 uint16_t* unpack_voice_SLQSVoiceManageCalls_t::pFailCause

8.691 unpack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID

8.691.1 Detailed Description

This structure contains the flash information associated with a call.

Parameters

pCallID	• Unique call identifier associated with the current call.

8.691.2 Field Documentation

8.691.2.1 uint8_t* unpack_voice_SLQSVoiceSendFlash_t::pCallID

8.692 unpack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference

Data Fields

- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.692.1 Detailed Description

This structure contains Voice Set Call Barring Password Unpack Parameters

Parameters

pFailCause	• Supplementary services failure cause (optional) • see qaGobiApiTableVoiceCallEndReasons.h for more information. • 0xFFFF,if Not Available
pAlphaIDInfo	• Pointer to structure of alphaIDInfo (optional) – See voice_alphaIDInfo for more information
pCCResType	• Call Control Result Type (optional) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - Not Available
pCallID	• Call ID of the voice call that resulted from call control. (optional) • It is present when pCCResType is present and is Voice. • If zero(0) then invalid.
pCCSUPSType	• Supplementary service data that resulted from call control (optional) • Data is present when pCCResultType is present and is other than Voice. – See voice_ccSUPSType for more information

Note

Using NULL for the pointers would make sure that the parameter is not returned or has default value.

8.692.2 Field Documentation

8.692.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pAlphaIDInfo

8.692.2.2 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCallID

8.692.2.3 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCResType

8.692.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCSUPSType

8.692.2.5 uint16_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pFailCause

8.693 unpack_voice_SLQSVoiceSetConfig_t Struct Reference

Data Fields

- uint8_t * pAutoAnsStatus
- uint8_t * pAirTimerStatus
- uint8_t * pRoamTimerStatus
- uint8_t * pTTYConfigStatus
- uint8_t * pPrefVoiceSOStatus
- uint8_t * pVoiceDomainPrefStatus

8.693.1 Detailed Description

This structure contains information about the Set Configuration Response Parameters.

Parameters

pAutoAnsStatus	• Auto Answer Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
pAirTimerStatus	• Air Timer Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
pRoamTimer-Status	• Roam Timer Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.

pTTYConfig-Status	• TTY Config Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
pPrefVoiceSO-Status	• Preferred Voice SO Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.
pVoiceDomain-PrefStatus	• Voice-Domain Preference Status. (optional) • Values: – 0x00 - Information was written successfully – 0x01 - Information write failed – 0xFF - Not Available.

Note

Parameters which are mentioned as NULL will be ignored.

8.693.2 Field Documentation

8.693.2.1 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAirTimerStatus

8.693.2.2 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAutoAnsStatus

8.693.2.3 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSOStatus

8.693.2.4 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pRoamTimerStatus

8.693.2.5 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pTTYConfigStatus

8.693.2.6 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pVoiceDomainPrefStatus

8.694 unpack_voice_SLQSVoiceSetSUPSService_t Struct Reference

Data Fields

- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResultType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.694.1 Detailed Description

This structure contains Supplementary Service response parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

pFailCause	• Supplementary service failure causes (optional, supply NULL if not required). – 0xFFFF is the value when the information is not received from device
pAlphaIDInfo	• Pointer to structure of alphaIDInfo. The parameter used to pass the alpha (if any) given by the SIM/R-UIM after call control (optional, supply NULL if not required) – See voice_alphaIDInfo for more information
pCCResultType	• Call control result types (optional, supply NULL if not required) – 0x00 - CC_RESULT_TYPE_VOICE - Voice – 0x01 - CC_RESULT_TYPE_SUPS - Supplementary service – 0x02 - CC_RESULT_TYPE_USSD - Unstructured supplementary service – 0xFF - if the device does not provide this information
pCallID	• Unique call identifier for the dialed call (optional, supply NULL if not required) – 0x00 - if the device does not provide this information
pCCSUPSType	• Data is present when pCCResultType is present and is other than Voice. (optional, supply NULL if not required) – See voice_ccSUPSType for more information

8.694.2 Field Documentation

8.694.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetSUPSService_t::pAlphaIDInfo

8.694.2.2 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCallID

8.694.2.3 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCCResultType

8.694.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceSetSUPSService_t::pCCSUPSType

8.694.2.5 uint16_t* unpack_voice_SLQSVoiceSetSUPSService_t::pFailCause

8.695 unpack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID

8.695.1 Detailed Description

This structure contains parameters of continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Start continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device

8.695.2 Field Documentation

8.695.2.1 uint8_t* unpack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.696 unpack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID

8.696.1 Detailed Description

This structure contains parameters of stop continuous DTMF

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. Stop continuous DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device

8.696.2 Field Documentation

8.696.2.1 uint8_t unpack_voice_SLQSVoiceStopContDTMF_t::callID

8.697 unpack_wds_DHCPv4ClientLease_ind_t Struct Reference

Data Fields

- wds_DHCPProfileIdTlv ProfileIdTlv
- wds_DHCPLeaseStateTlv DHCPv4LeaseStateTlv
- wds_IPv4AdTlv IPv4AddrTlv
- wds_DHCPLeaseOptTlv DHCPv4LeaseOptTlv

8.697.1 Detailed Description

This structure contains DHCP IPv4 client lease information

Parameters

ProfileIdTlv	profile identifier information, see wds_DHCPProfileIdTlv for more details
DHCPv4Lease-StateTlv	lease state, see wds_DHCPLeaseStateTlv for more details
IPv4AddrTlv	IPv4 address, see wds_IPv4AdTlv for more details
DHCPv4Lease-OptTlv	DHCP lease option, see wds_DHCPLeaseOptTlv for more details

8.697.2 Field Documentation

8.697.2.1 wds_DHCPLeaseOptTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseOptTlv

8.697.2.2 wds_DHCPLeaseStateTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseStateTlv

8.697.2.3 wds_IPv4AdTlv unpack_wds_DHCPv4ClientLease_ind_t::IPv4AddrTlv

8.697.2.4 wds_DHCPProfileIdTlv unpack_wds_DHCPv4ClientLease_ind_t::ProfileIdTlv

8.698 unpack_wds_GetAutoconnect_t Struct Reference

Data Fields

- uint32_t * psetting

8.698.1 Detailed Description

auto connect data session setting parameter.

Parameters

pSetting	• NDIS auto connect setting – 0 - Disabled – 1 - Enabled

8.698.2 Field Documentation

8.698.2.1 uint32_t* unpack_wds_GetAutoconnect_t::psetting

8.699 unpack_wds_GetByteTotals_t Struct Reference

Data Fields

- uint64_t * pTXTotalBytes
- uint64_t * pRXTotalBytes

8.699.1 Detailed Description

Parameters

pTXTotalBytes	• Bytes transmitted without error
pRXTotalBytes	• Bytes received without error

8.699.2 Field Documentation

8.699.2.1 uint64_t* unpack_wds_GetByteTotals_t::pRXTotalBytes

8.699.2.2 uint64_t* unpack_wds_GetByteTotals_t::pTXTotalBytes

8.700 unpack_wds_GetConnectionRate_t Struct Reference

Data Fields

- uint32_t currentChannelTXRate
- uint32_t currentChannelRXRate
- uint32_t maxChannelTXRate
- uint32_t maxChannelRXRate

8.700.1 Detailed Description

Parameters

pCurrentChannelTXRate	• Current channel Tx rate (in bps)
pCurrentChannelRXRate	• Current channel Rx rate (in bps)
pMaxChannelTXRate	• Maximum Tx rate (bps) that may be assigned to device by serving system.
pMaxChannelRXRate	• Maximum Rx rate (bps) that may be assigned to device by serving system.

8.700.2 Field Documentation

8.700.2.1 uint32_t unpack_wds_GetConnectionRate_t::currentChannelRXRate

8.700.2.2 uint32_t unpack_wds_GetConnectionRate_t::currentChannelTXRate

8.700.2.3 uint32_t unpack_wds_GetConnectionRate_t::maxChannelRXRate

8.700.2.4 uint32_t unpack_wds_GetConnectionRate_t::maxChannelTXRate

8.701 unpack_wds_GetDataBearerTechnology_t Struct Reference

Data Fields

- uint32_t * pDataBearer

8.701.1 Detailed Description

Parameters

pDataBearer[OUT]	• Data bearer technology – 0x01 - CDMA2000 1x – 0x02 - CDMA 1xEV-DO Rev 0 – 0x03 - GSM – 0x04 - UMTS – 0x05 - CDMA2000 HRPD (1xEV-DO Rev A) – 0x06 - EDGE – 0x07 - HSDPA AND WCDMA – 0x08 - WCDMA AND HSUPA – 0x09 - HSDPA AND HSUPA – 0x0A - LTE – 0x0B - CDMA2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown

8.701.2 Field Documentation

8.701.2.1 uint32_t* unpack_wds_GetDataBearerTechnology_t::pDataBearer

8.702 unpack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t pdptype
- uint32_t ipaddr
- uint32_t pridns
- uint32_t secdns
- uint16_t ipaddrv6
- uint16_t pridnsv6
- uint16_t secdnsv6
- uint32_t auth
- uint8_t namesize

- int8_t name [255]
- uint8_t apnsize
- int8_t apnname [255]
- uint8_t usersize
- int8_t username [255]

8.702.1 Detailed Description

Reads the default profile settings from the device. The default profile is used to establish an auto connect data session.

Parameters

pdptype	• Packet Data Protocol (PDP) type specifies the type of data payload exchanged over the air link when the packet data session is established with this profile – 0 - PDP-IP (IPv4) – 1 - PDP-PPP – 2 - PDP-IPv6 – 3 - PDP-IPv4v6
ipaddr	• Preferred IPv4 address to be assigned to device
pridns	• Primary DNS IPv4 address preference
secdns	• Secondary DNS IPv4 address preference
ipaddrv6	• Preferred IPv6 address to be assigned to device
pridnsv6	• Primary DNS Pv6 address preference
secdnsv6	• Secondary DNS IPv6 address preference
auth	• Bitmap that indicates authentication algorithm preference – 0x00000001 - PAP preference * 0 - Never performed * 1 - May be performed – 0x00000002 - CHAP preference * 0 - Never performed * 1 - May be performed – All other bits are reserved and must be set to 0 – If more than 1 bit is set, then device decides which authentication procedure is performed while setting up data session e.g. the device may have a policy to select the most secure authentication mechanism.
namesize	• Maximum number of characters (including NULL terminator) that profile name array can contain.

name	• Profile name
apnsize	• Maximum number of characters (including NULL terminator) that APN name array can contain
apnname	• Access point name. NULL-terminated string parameter that is a logical name used to select GGSN and external packet data network. • If value is NULL or omitted, then subscription default value will be requested.
usersize	• Maximum number of characters (including NULL terminator) that username array can contain.
username	• Username used during network authentication

8.702.2 Field Documentation

8.702.2.1 int8_t unpack_wds_GetDefaultProfile_t::apnname[255]

8.702.2.2 uint8_t unpack_wds_GetDefaultProfile_t::apnsize

8.702.2.3 uint32_t unpack_wds_GetDefaultProfile_t::auth

8.702.2.4 uint32_t unpack_wds_GetDefaultProfile_t::ipaddr

8.702.2.5 uint16_t unpack_wds_GetDefaultProfile_t::ipaddrv6

8.702.2.6 int8_t unpack_wds_GetDefaultProfile_t::name[255]

8.702.2.7 uint8_t unpack_wds_GetDefaultProfile_t::namesize

8.702.2.8 uint32_t unpack_wds_GetDefaultProfile_t::pdptype

8.702.2.9 uint32_t unpack_wds_GetDefaultProfile_t::pridns

8.702.2.10 uint16_t unpack_wds_GetDefaultProfile_t::pridnsv6

8.702.2.11 uint32_t unpack_wds_GetDefaultProfile_t::secdns

8.702.2.12 uint16_t unpack_wds_GetDefaultProfile_t::secdnsv6

8.702.2.13 int8_t unpack_wds_GetDefaultProfile_t::username[255]

8.702.2.14 uint8_t unpack_wds_GetDefaultProfile_t::usersize

8.703 unpack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t index

8.703.1 Detailed Description

Parameters

index	profile index

8.703.2 Field Documentation

8.703.2.1 uint8_t unpack_wds_GetDefaultProfileNum_t::index

8.704 unpack_wds_GetDormancyState_t Struct Reference

Data Fields

- uint32_t dormancyState

8.704.1 Detailed Description

Parameters

dormancyState	• Dormancy state of current packet data session – 1 - Traffic channel dormant – 2 - Traffic channel active

8.704.2 Field Documentation

8.704.2.1 uint32_t unpack_wds_GetDormancyState_t::dormancyState

8.705 unpack_wds_GetLastMobileIPError_t Struct Reference

Data Fields

- uint32_t error

8.705.1 Detailed Description

structure to store last mobile IP error.

Parameters

error	• Status of last MIP call (or attempt) – Zero - Success – NonZero - Error code See qaGobiApiTableCallEndReasons.h for Mobile IP Error codes

8.705.2 Field Documentation

8.705.2.1 uint32_t unpack_wds_GetLastMobileIPError_t::error

8.706 unpack_wds_GetMobileIP_t Struct Reference

Data Fields

- uint32_t mipMode

8.706.1 Detailed Description

Parameters

mipMode	• Mobile IP setting – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only

8.706.2 Field Documentation

8.706.2.1 uint32_t unpack_wds_GetMobileIP_t::mipMode

8.707 unpack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t enabled
- uint32_t address
- uint32_t primaryHA
- uint32_t secondaryHA
- uint8_t revTunneling
- uint8_t naiSize
- int8_t NAI [255]
- uint32_t HASPI
- uint32_t AAASPI
- uint32_t HAState
- uint32_t AAAState

8.707.1 Detailed Description

Parameters

enabled	• Profile enabled: – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
address	• Home IPv4 address: – 0xFFFFFFFF - Unknown

primaryHA	• Primary home agent IPv4 address – 0xFFFFFFFF - Unknown
secondaryHA	• Secondary home agent IPv4 address – 0xFFFFFFFF - Unknown
revTunneling	• Reverse tunneling enabled – 0 - Disabled – 1 - Enabled – 0xFF - Unknown
naiSize	• The maximum number of characters (including NULL terminator) that the NAI array can contain.
NAI	• Network access identifier string
HASPI	• Home agent security parameter index
AAASPI	• AAA server security parameter index – 0xFFFFFFFF - Unknown
HAState	• Home agent key state – 0 - Unset – 1 - Set, default value – 2 - Set, modified from default – 3 - 0xFFFFFFFF - Unknown
AAAState	• AAA key state – 0 - Unset – 1 - Set, default value – 2 - Set, modified from default – 3 - 0xFFFFFFFF - Unknown

8.707.2 Field Documentation

8.707.2.1 uint32_t unpack_wds_GetMobileIPProfile_t::AAASPI

8.707.2.2 uint32_t unpack_wds_GetMobileIPProfile_t::AAAState

8.707.2.3 uint32_t unpack_wds_GetMobileIPProfile_t::address

8.707.2.4 uint8_t unpack_wds_GetMobileIPProfile_t::enabled

8.707.2.5 uint32_t unpack_wds_GetMobileIPProfile_t::HASPI

8.707.2.6 uint32_t unpack_wds_GetMobileIPProfile_t::HAState

8.707.2.7 int8_t unpack_wds_GetMobileIPProfile_t::NAI[255]

8.707.2.8 uint8_t unpack_wds_GetMobileIPProfile_t::naiSize

8.707.2.9 uint32_t unpack_wds_GetMobileIPProfile_t::primaryHA

8.707.2.10 uint8_t unpack_wds_GetMobileIPProfile_t::revTunneling

8.707.2.11 uint32_t unpack_wds_GetMobileIPProfile_t::secondaryHA

8.708 unpack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * pTXPacketSuccesses
- uint32_t * pRXPacketSuccesses
- uint32_t * pTXPacketErrors
- uint32_t * pRXPacketErrors
- uint32_t * pTXPacketOverflows
- uint32_t * pRXPacketOverflows
- uint64_t * pTXOkBytesCount
- uint64_t * pRXOkBytesCount
- uint64_t * pTXOKBytesLastCall
- uint64_t * pRXOKBytesLastCall
- uint32_t * pTXDroppedCount
- uint32_t * pRXDroppedCount

8.708.1 Detailed Description

Parameters

pTXPacket-Successes	• No of transmitted Packets without error.
pRXPacket-Successes	• No of received Packets without error.
pTXPacketErrors	• Number of outgoing packets with framing errors.
pRXPacket-Errors	• Number of incoming packets with framing errors.
pTXPacket-Overflows	• Number of packets dropped because Tx buffer overflowed (out of memory).
pRXPacket-Overflows	• Number of packets dropped because Rx buffer overflowed (out of memory).

pTXOkBytes-Count	• Number of bytes transmitted without error.
pRXOkBytes-Count	• Number of bytes received without error.
pTXOKBytes-LastCall	• Number of bytes transmitted without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support
pRXOKBytes-LastCall	• Number of bytes received without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support
pTXDropped-Count	• Number of outgoing packets dropped.
pRXDropped-Count	• Number of incoming packets dropped.

8.708.2 Field Documentation

8.708.2.1 uint32_t* unpack_wds_GetPacketStatistics_t::pRXDroppedCount

8.708.2.2 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOkBytesCount

8.708.2.3 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOKBytesLastCall

8.708.2.4 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketErrors

8.708.2.5 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketOverflows

8.708.2.6 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketSuccesses

8.708.2.7 uint32_t* unpack_wds_GetPacketStatistics_t::pTXDroppedCount

8.708.2.8 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOkBytesCount

8.708.2.9 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOKBytesLastCall

8.708.2.10 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketErrors

8.708.2.11 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketOverflows

8.708.2.12 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketSuccesses

8.709 unpack_wds_GetPacketStatus_t Struct Reference

Data Fields

- uint32_t tXPacketSuccesses
- uint32_t rXPacketSuccesses
- uint32_t tXPacketErrors
- uint32_t rXPacketErrors
- uint32_t tXPacketOverflows
- uint32_t rXPacketOverflows
- uint64_t tXOkBytesCount
- uint64_t rXOkBytesCount
- uint64_t tXOKBytesLastCall
- uint64_t rXOKBytesLastCall
- uint32_t tXDroppedCount
- uint32_t rXDroppedCount

8.709.1 Detailed Description

Returns the packet data transfer statistics since the start of the current packet data.

Parameters

tXPacket-Successes	• No. of packets transmitted without error
rXPacket-Successes	• No. of packets received without error
tXPacketErrors	• No. of outgoing packets with framing errors
rXPacketErrors	• No. of incoming packets with framing errors
txPacket-Overflows	• Number of packets dropped because Tx buffer overflowed
rxPacket-Overflows	• Number of packets dropped because Rx buffer overflowed
txOkBytesCount	• No of bytes transmitted without error.
rxOkBytesCount	• No of bytes received without error.
txOKBytesLast-Call	• No of bytes transmitted without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support
rxOKBytesLast-Call	• Number of bytes received without error during the last data call (0 if no call was made earlier). Returned only if not in a call, and when the previous call was made using RmNet (for any devices that support

txDroppedCount	• Number of outgoing packets dropped.
rxDroppedCount	• Number of incoming packets dropped.

8.709.2 Field Documentation

8.709.2.1 uint32_t unpack_wds_GetPacketStatus_t::rXDroppedCount

8.709.2.2 uint64_t unpack_wds_GetPacketStatus_t::rXOkBytesCount

8.709.2.3 uint64_t unpack_wds_GetPacketStatus_t::rXOKBytesLastCall

8.709.2.4 uint32_t unpack_wds_GetPacketStatus_t::rXPacketErrors

8.709.2.5 uint32_t unpack_wds_GetPacketStatus_t::rXPacketOverflows

8.709.2.6 uint32_t unpack_wds_GetPacketStatus_t::rXPacketSuccesses

8.709.2.7 uint32_t unpack_wds_GetPacketStatus_t::tXDroppedCount

8.709.2.8 uint64_t unpack_wds_GetPacketStatus_t::tXOkBytesCount

8.709.2.9 uint64_t unpack_wds_GetPacketStatus_t::tXOKBytesLastCall

8.709.2.10 uint32_t unpack_wds_GetPacketStatus_t::tXPacketErrors

8.709.2.11 uint32_t unpack_wds_GetPacketStatus_t::tXPacketOverflows

8.709.2.12 uint32_t unpack_wds_GetPacketStatus_t::tXPacketSuccesses

8.710 unpack_wds_GetSessionDuration_t Struct Reference

Data Fields

- uint64_t callDuration

8.710.1 Detailed Description

Parameters

callDuration	• Duration of the current packet session in milliseconds

8.710.2 Field Documentation

8.710.2.1 uint64_t unpack_wds_GetSessionDuration_t::callDuration

8.711 unpack_wds_GetSessionState_t Struct Reference

Data Fields

- uint32_t connectionStatus

8.711.1 Detailed Description

Parameters

connection-Status	state of the current packet data session

8.711.2 Field Documentation

8.711.2.1 uint32_t unpack_wds_GetSessionState_t::connectionStatus

8.712 unpack_wds_RMSetTransferStatistics_t Struct Reference

8.712.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.713 unpack_wds_SetMobileIPProfile_t Struct Reference

8.713.1 Detailed Description

Parameters

NULL	• this is a dummy structure

8.714 unpack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- PackCreateProfileOut * pCreateProfileOut
- uint8_t * pProfileID
- uint16_t Tlvresult

8.714.1 Detailed Description

Parameters

pCreateProfile-Out	• SLQS Create profile Information

pProfileID	• SLQS profile identifier information
TlvresuIt	• TLV present flag, TURE if presented, otherwise FALSE

8.714.2 Field Documentation

8.714.2.1 PackCreateProfileOut* unpack_wds_SLQSCreateProfile_t::pCreateProfileOut

8.714.2.2 uint8_t* unpack_wds_SLQSCreateProfile_t::pProfileID

8.714.2.3 uint16_t unpack_wds_SLQSCreateProfile_t::TlvresuIt

8.715 unpack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint16_t extendedErrorCode

8.715.1 Detailed Description

Parameters

pExtendedError-Code[OUT]	• The extended error code received from DS Profile subsystem of type eWDS_ERR_PROFILE_-REG_xxx. • Error code will only will be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERN-AL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.

8.715.2 Field Documentation

8.715.2.1 uint16_t unpack_wds_SLQSDeleteProfile_t::extendedErrorCode

8.716 unpack_wds_SLQSGet3GPPConfigItem_t Struct Reference

Data Fields

- uint16_t LTEAttachProfile
- uint16_t profileList [5]
- uint8_t defaultPDNEnabled
- uint8_t _3gppRelease
- uint16_t LTEAttachProfileList [24]
- uint16_t LTEAttachProfileListLen

8.716.1 Detailed Description

Parameters

	pLTEAttach-Profile	• Optional parameter • LTE Attach Profile – points to a single WORD Value indicating the attached LTE Profile – Optional parameter with possible values 1-16 (EM/MC73xx or earlier) • This setting is deprecated on MC/EM74xx
	profileList	Profile List • an array of 4 profile configurations • Each element points to a single WORD value indicating profile • Optional parameter with possible values – 1 - 16 (MC/EM73xx and before) – 1 - 24 (MC/EM74xx and onwards) • function SLQSGet3GPPConfigItem() returns a default value 255 if no 3gpp configuration is present • Note: the 5th entry is currently ignored, please set it to zero
out	*defaultPDN-Enabled*	• 0 - disabled • 1 - enabled
out	*_3gppRelease*	3GPP release • 0 - Release_99 • 1 - Release_5 • 2 - Release_6 • 3 - Release_7 • 4 - Release_8 • 5 - Release_9 (In 9x30 and toworads) • 6 - Release_10 (In 9x30 and toworads) • 7 - Release_11 (In 9x30 and toworads)
out	*LTEAttach-ProfileList*	• pointer to WORD array indiciating LTE Attach Profile List – Optional parameter – possible values: 1-24 – This setting is only supported for MC/EM74xx onwards – Please provide attach profiles in order of decreasing priority in this list.
in,out	*LTEAttach-ProfileListLen*	• Number of element in pLTEAttachProfileList – valid range: 1-24 – This setting is only supported for MC/EM74xx onwards

8.716.2 Field Documentation

8.716.2.1 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::_3gppRelease

8.716.2.2 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::defaultPDNEnabled

8.716.2.3 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfile

8.716.2.4 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileList[24]

8.716.2.5 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileListLen

8.716.2.6 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::profileList[5]

8.717 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

Data Fields

- uint8_t prefNetwork
- uint8_t networkInfoLen
- currNetworkInfo currNetworkInfo [255]

8.717.1 Detailed Description

Data System Status

Parameters

prefNetwork	• Preferred Network • Values: – 0 - 3GPP – 1 - 3GPP2
networkInfoLen	• As input parameter size assigned to next parameter i.e. network information • As output the actual number of network information elements returned by the device
currNetworkInfo	• Network information • See currNetworkInfo for more details

8.717.2 Field Documentation

8.717.2.1 currNetworkInfo unpack_wds_SLQSGetCurrDataSystemStat_t::currNetworkInfo[255]

8.717.2.2 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::networkInfoLen

8.717.2.3 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::prefNetwork

8.718 unpack_wds_SLQSGetCurrentChannelRate_t Struct Reference

Data Fields

- uint32_t current_channel_tx_rate
- uint32_t current_channel_rx_rate
- uint32_t max_channel_tx_rate
- uint32_t max_channel_rx_rate

8.718.1 Detailed Description

Parameters

current_channel-_tx_rate	• Current Channel Tx Rate. • Instantaneous channel Tx rate in bits per second. • In 9x15, this is the total current channel rate for all PDNs combined. • In 9x30 and later, this is the channel rate for a specific PDN.
current_channel-_rx_rate	• Current Channel Rx Rate. • Instantaneous channel Rx rate in bits per second. • In 9x15, this is the total current channel rate for all PDNs combined. • In 9x30 and later, this is the channel rate for a specific PDN
max_channel_-tx_rate	• Max Channel Tx Rate. • Maximum total Tx rate that modem is able to support in current serving system in bits per second. • In 9x15, this is a default hard coded value for the current serving system.
max_channel_-rx_rate	• Max Channel Rx Rate. • Maximum total Rx rate that modem is able to support in current serving system in bits per second. • In 9x15, this is a default hard coded value for the current serving system.

8.718.2 Field Documentation

8.718.2.1 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_rx_rate

8.718.2.2 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_tx_rate

8.718.2.3 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_rx_rate

8.718.2.4 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_tx_rate

8.719 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference

Data Fields

- uint8_t dataBearerMask
- qmiWDSDataBearerTechnology curDataBearerTechnology
- qmiWDSDataBearerTechnology lastCallDataBearerTechnology

8.719.1 Detailed Description

Structure to hold the data bearer technology values

Parameters

dataBearerMask[OUT]	• This bit mask indicates if data bearer information for the current and/or last call has been received from the device. If a bit is set, then the information is available in the corresponding structure i.e. the one provided by the caller. Refer to qmiDataBearerMasks for bit-mask positions.
curDataBearerTechnology[OUT]	• current data bearer technology value. – NULL if the parameter is not required
lastCallDataBearerTechnology[OUT]	• last call data bearer technology value. – NULL if the parameter is not required

8.719.2 Field Documentation

8.719.2.1 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::curDataBearerTechnology

8.719.2.2 uint8_t unpack_wds_SLQSGetDataBearerTechnology_t::dataBearerMask

8.719.2.3 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::lastCallDataBearerTechnology

8.720 unpack_wds_SLQSGetDUNCallInfo_t Struct Reference

Data Fields

- connectionStatus connectionStatus
- uint16_t callEndReason
- uint64_t txOKBytesCount
- uint64_t rxOKBytesCount
- uint8_t dormancyStatus
- uint8_t dataBearerTech
- dunchannelRate channelRate
- uint64_t lastCallTXOKBytesCnt
- uint64_t lastCallRXOKBytesCnt
- uint64_t mdmCallDurationActive
- uint8_t lastCallDataBearerTech

8.720.1 Detailed Description

This structure contains the DUN Call Info response parameters

Parameters

connectionStatus	• See connectionStatus for more information
callEndReason	• Last modem call end reason • See qaGobiApiTableCallEndReasons.h for Call End Reason • Only valid if the last call made was DUN, else zero is returned

txOKBytesCount	• Number of bytes transmitted without error • Returned only if a data call is up
rxOKBytesCount	• Number of bytes received without error • Returned only if a data call is up
dormancyStatus	• Current traffic channel status • Returned if a data call is up – 0x01 - Traffic channel dormant – 0x02 - Traffic channel active
dataBearerTech	• Current data bearer technology • Returned only if a data call is up – 0x01 - cdma2000 1X – 0x02 - cdma2000 HRPD (1xEV-DO) – 0x03 - GSM – 0x04 - UMTS – 0x05 - cdma200 HRPD (1xEV-DO RevA) – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - cdma2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown
channelRate	• See dunchannelRate for more information
lastCallTXOK-BytesCnt	• Number of bytes transmitted without error during the last data call (0 if no call was made). • Return only if not in a call and the previous call was made using DUN.
lastCallRXOK-BytesCnt	• Number of bytes transmitted without error during the last data call (0 if no call was made). • Return only if not in a call and the previous call was made using DUN.

mdmCall-DurationActive	• Duration that the call is active in milliseconds • If the modem connection status is connected, this represents the active duration of the current DUN call • If the modem connection status is disconnected, this represents the duration of the last DUN call since the device was powered up (0 if no call has been made or if the last call was not DUN)
lastCallData-BearerTech	• Last Call Data Bearer Technology • Returned only if not in a call and when the previous call was made using DUN – 0x01 - cdma2000 1X – 0x02 - cdma2000 HRPD (1xEV-DO) – 0x03 - GSM – 0x04 - UMTS – 0x05 - cdma200 HRPD (1xEV-DO Rev A) – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - WCDMA and HSUPA – 0x09 - HSDPA and HSUPA – 0x0A - LTE – 0x0B - cdma2000 EHRPD – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA – 0x10 - HSDPA+ and 64QAM – 0x11 - HSDPA+, 64QAM and HSUPA – 0x12 - TDSCDMA – 0x13 - TDSCDMA and HSDPA – 0xFF - Unknown

8.720.2 Field Documentation

8.720.2.1 uint16_t unpack_wds_SLQSGetDUNCallInfo_t::callEndReason

8.720.2.2 dunchannelRate unpack_wds_SLQSGetDUNCallInfo_t::channelRate

8.720.2.3 connectionStatus unpack_wds_SLQSGetDUNCallInfo_t::connectionStatus

8.720.2.4 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dataBearerTech

8.720.2.5 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dormancyStatus

8.720.2.6 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallDataBearerTech

8.720.2.7 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallRXOKBytesCnt

8.720.2.8 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallTXOKBytesCnt

8.720.2.9 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::mdmCallDurationActive

8.720.2.10 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::rxOKBytesCount

8.720.2.11 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::txOKBytesCount

8.721 unpack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- UnPackGetProfileSettingOut * pProfileSettings
- uint8_t ProfileType
- uint16_t Tlvresult

8.721.1 Detailed Description

This structure contains the profile setting information of parameter pOutput for API unpack_wds_SLQSGetProfile-Settings

Parameters

SlqsProfile3GPP	3GPP profile
SlqsProfile3GP-P2	3GPP2 profile

8.721.2 Field Documentation

8.721.2.1 UnPackGetProfileSettingOut* unpack_wds_SLQSGetProfileSettings_t::pProfileSettings

8.721.2.2 uint8_t unpack_wds_SLQSGetProfileSettings_t::ProfileType

8.721.2.3 uint16_t unpack_wds_SLQSGetProfileSettings_t::Tlvresult

8.722 unpack_wds_SLQSGetRuntimeSettings_t Struct Reference

Data Fields

- uint32_t IPv4
- uint8_t ProfileName [128]
- uint32_t PDPType
- uint8_t APNName [128]
- uint32_t PrimaryDNSV4
- uint32_t SecondaryDNSV4
- LibPackUMTSQoS UMTSGrantedQoS
- struct wds_GPRSQoS GPRSGrantedQoS
- uint8_t Username [128]
- uint32_t Authentication
- struct wds_ProfileIdentifier ProfileID
- uint32_t GWAddressV4
- uint32_t SubnetMaskV4
- uint8_t PCSCFAddrPCO
- struct wds_PCSCFIPv4ServerAddressList ServerAddrList
- struct wds_PCSCFFQDNAddressList PCSCFFQDNAddrList
- uint16_t PrimaryDNSV6 [8]

- uint16_t SecondaryDNSV6 [8]
- uint32_t Mtu
- struct wds_DomainNameList DomainList
- uint8_t IPFamilyPreference
- uint8_t IMCNflag
- uint16_t Technology
- struct wds_IPV6AddressInfo IPV6AddrInfo
- struct wds_IPV6GWAddressInfo IPV6GWAddrInfo

8.722.1 Detailed Description

Parameters

IPv4	ipv4 address
ProfileName	profile name
PDPType	PDP type
APNName	APN name
PrimaryDNSV4	primary dns IPV4
SecondaryDNS-V4	secondary dns IPV4
UMTSGranted-QoS	UMTS Granted Qos
GPRSGranted-QoS	GPRS Granted QoS
Username	username for authentication process
Authentication	authentication for authentication process
ProfielID	profile ID
GWAddressV4	Gateway IPv4
SubnetMaskV4	Subnet mask IPV4
PCSCFAddrPC-O	PCSCF address PCO
ServerAddrList	PCSCF server address list IPV4
PCSCFFQDN-AddrList	PCSCF FQDN address list IPV4
PrimaryDNSV6	Primary DNS IPV6
SecondryDNS-V6	Secondry DNS IPV6
Mtu	Maximum Transfer Unit
DomainList	domain list
IPFamily-Preference	ip family preference, it could be IPV4 or IPV6
IMCNflag	IM control flag, value: TRUE or FALSE
Technology	technology on which current packet data session is in progress • Values: – 32767 - CDMA – 32764 - UMTS – 30592 - EPC – 30584 - modem link local
IPV6AddrInfo	address information IPV6
IPV6GWAddr-Info	gateway address information IPV6

8.722.2 Field Documentation

8.722.2.1 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::APNName[128]

8.722.2.2 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Authentication

8.722.2.3 struct wds_DomainNameList unpack_wds_SLQSGetRuntimeSettings_t::DomainList

8.722.2.4 struct wds_GPRSQoS unpack_wds_SLQSGetRuntimeSettings_t::GPRSGrantedQoS

8.722.2.5 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::GWAddressV4

8.722.2.6 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IMCNflag

8.722.2.7 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IPFamilyPreference

8.722.2.8 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::IPv4

8.722.2.9 struct wds_IPV6AddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPV6AddrInfo

8.722.2.10 struct wds_IPV6GWAddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPV6GWAddrInfo

8.722.2.11 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Mtu

8.722.2.12 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::PCSCFAddrPCO

8.722.2.13 struct wds_PCSCFFQDNAddressList unpack_wds_SLQSGetRuntimeSettings_t::PCSCFFQDNAddrList

8.722.2.14 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PDPType

8.722.2.15 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV4

8.722.2.16 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV6[8]

8.722.2.17 struct wds_ProfileIdentifier unpack_wds_SLQSGetRuntimeSettings_t::ProfileID

8.722.2.18 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::ProfileName[128]

8.722.2.19 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV4

8.722.2.20 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV6[8]

8.722.2.21 struct wds_PCSCFIPv4ServerAddressList unpack_wds_SLQSGetRuntimeSettings_t::ServerAddrList

8.722.2.22 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SubnetMaskV4

8.722.2.23 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::Technology

8.722.2.24 LibPackUMTSQoS unpack_wds_SLQSGetRuntimeSettings_t::UMTSGrantedQoS

8.722.2.25 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::Username[128]

8.723 unpack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint16_t * pExtErrorCode

8.723.1 Detailed Description

This structure contains out parameters for unpack_wds_SLQSModifyProfile

Parameters

pExtErrorCode	• The extended error code received from DS Profile subsystem of type eWDS_ERR_PROFILE_-REG_xxx. • Error code will only will be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERN-AL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.

8.723.2 Field Documentation

8.723.2.1 uint16_t* unpack_wds_SLQSModifyProfile_t::pExtErrorCode

8.724 unpack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.724.1 Detailed Description

Parameters

Tlvresult	unpack result

8.724.2 Field Documentation

8.724.2.1 uint16_t unpack_wds_SLQSSetIPFamilyPreference_t::Tlvresult

8.725 unpack_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference

Data Fields

- uint8_t conn_status
- uint8_t reconfigReqd
- uint16_t sessionEndReason
- uint16_t verboseSessnEndReasonType
- uint16_t verboseSessnEndReason
- uint8_t ipFamily
- uint16_t techName
- uint8_t bearerID

8.725.1 Detailed Description

Parameters

conn_status	connection status
reconfigReqd	Indicates whether the network interface on the host needs to be reconfigured.
sessionEnd-Reason	Call End Reason
verboseSessn-EndReasonType	Verbose call end reason type
verboseSessn-EndReason	Reason the call ended (verbose)
ipFamily	IP family of the packet data connection.
techName	Technology name of the packet data connection.
bearerID	• bearer ID (3GPP) or RLP ID (3GPP2) of the packet data connection. • Valid Values - 0 to 16 • 0xFF - Invalid value.

8.725.2 Field Documentation

8.725.2.1 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::bearerID

8.725.2.2 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::conn_status

8.725.2.3 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::ipFamily

8.725.2.4 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::reconfigReqd

8.725.2.5 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::sessionEndReason

8.725.2.6 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::techName

8.725.2.7 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReason

8.725.2.8 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReasonType

8.726 unpack_wds_SLQSSetWdsEventCallback_ind_t Struct Reference

Data Fields

- uint8_t xferStatAvail
- uint64_t tx_bytes
- uint64_t rx_bytes
- uint64_t tx_pkts
- uint64_t rx_pkts
- uint8_t mipstatAvail
- uint32_t mipStatus
- uint8_t dBTechAvail
- uint32_t dBTechnology
- uint8_t dormancyStatAvail
- uint32_t dormancyStatus
- uint8_t currDBTechAvail
- uint32_t ratMask
- uint32_t soMask
- uint8_t dataSysStatAvail

- uint8_t prefNetwork
- uint8_t netInfoLen
- wds_currNetworkInfo currNWInfo [255]
- uint8_t dBtechExtAvail
- uint32_t dBtechnologyExt
- uint32_t dBTechExtRatValue
- uint64_t dBTechExtSoMask

8.726.1 Detailed Description

Parameters

xferStatAvail	transfer statistic available
tx_bytes	transmit bytes
rx_bytes	received bytes
tx_pkts	transmit packets
rx_pkts	received packets
mipstatAvail	Mobile IP status available
mipStatus	Mobile IP status
dBTechAvail	Data Bearer technology available
dBTechnology	Data Bearer technology
dormancyStatAvail	Dormancy status available
dormancyStatus	Dormancy status
currDBTechAvail	Current Data Bearer technology available
ratMask	RAT mask to indicate type of technology
soMask	SO mask to indicate the service type
dataSysStatAvail	Data System Status available
prefNetwork	preferred network
currNWInfo	Current Network Info
dBtechExtAvail	Data bearer technology extended available
dBtechnologyExt	Data bearer technology Extended.
dBTechExtRatValue	Data bearer technology Extended RAT Value
dBTechExtSoMask	Data bearer technology Extended SO Mask

8.726.2 Field Documentation

8.726.2.1 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::currDBTechAvail

8.726.2.2 wds_currNetworkInfo unpack_wds_SLQSSetWdsEventCallback_ind_t::currNWInfo[255]

8.726.2.3 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dataSysStatAvail

8.726.2.4 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechAvail

8.726.2.5 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechExtAvail

8.726.2.6 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtRatValue

8.726.2.7 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtSoMask

8.726.2.8 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechnology

8.726.2.9 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechnologyExt

8.726.2.10 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatAvail

8.726.2.11 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatus

8.726.2.12 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipstatAvail

8.726.2.13 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipStatus

8.726.2.14 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::netInfoLen

8.726.2.15 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::prefNetwork

8.726.2.16 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::ratMask

8.726.2.17 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_bytes

8.726.2.18 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_pkts

8.726.2.19 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::soMask

8.726.2.20 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_bytes

8.726.2.21 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_pkts

8.726.2.22 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::xferStatAvail

8.727 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4HwConfig * pHwConfig
- wdsDhcpv4OptionList * pRequestOptionList

8.727.1 Detailed Description

WDS SWI DHCPv4 Config Structure

Parameters

pHWConfig	• pointer to HW Config structure
pRequestOption-List	• pointer to Option List structure to be sent in DHCP request

8.727.2 Field Documentation

8.727.2.1 wdsDhcpv4HwConfig* unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pHwConfig

8.727.2.2 wdsDhcpv4OptionList* unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pRequestOptionList

8.728 unpack_wds_SLQSSGetLoopback_t Struct Reference

Data Fields

- uint8_t ByteLoopbackMode
- uint8_t ByteLoopbackMultiplier

8.728.1 Detailed Description

Parameters

ByteLoopback-Mode	• Loopback Mode. – 0 - Disable – 1 - Enable
ByteLoopback-Multiplier	• Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.728.2 Field Documentation

8.728.2.1 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMode

8.728.2.2 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMultiplier

8.729 unpack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint32_t * psid
- uint32_t * pFailureReason
- uint32_t * pVerboseFailReasonType
- uint32_t * pVerboseFailureReason

8.729.1 Detailed Description

Parameters

psid	• Assigned session ID when starting a data session
pFailureReason	• Reason data session failed to be established • See qaGobiApiTableCallEndReasons.h for Call End Reason
pVerboseFail-ReasonType	• Parameter describing type of verbose failure reason • See qaGobiApiTableCallEndReasons.h for Call End Reason Type
pVerboseFailure-Reason	• Verbose reason explaining why call failed. Depends on verbFailReasonType parameter • See qaGobiApiTableCallEndReasons.h for Call End Reason

8.729.2 Field Documentation

8.729.2.1 uint32_t* unpack_wds_SLQSStartDataSession_t::pFailureReason

8.729.2.2 uint32_t* unpack_wds_SLQSStartDataSession_t::psid

8.729.2.3 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailReasonType

8.729.2.4 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailureReason

8.730 unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference

Data Fields

- uint8_t contextId
- uint8_t bearerId
- int8_t apnName [100]
- uint32_t ipv4Address
- uint32_t ipv4GWAddress
- uint32_t prDNSIPv4Address
- uint32_t seDNSIPv4Address
- struct ipv6AddressInfo ipv6Address
- struct ipv6AddressInfo ipv6GWAddress
- uint16_t prDNSIPv6Address [8]
- uint16_t seDNSIPv6Address [8]
- uint32_t prPCSCFIPv4Address
- uint32_t sePCSCFIPv4Address
- uint16_t prPCSCFIPv6Address [8]
- uint16_t sePCSCFIPv6Address [8]

8.730.1 Detailed Description

This structure contains the unpack parameters retrieved by the API unpack_wds_SLQSWdsSwiPDPRuntime-Settings

Parameters

contextId	(optional) • Context Identifier – 0xFF - Not Available
bearerId	(optional) • Bearer Identity • An EPS bearer identity uniquely identifies an EPS bearer for one UE accessing via E-UTRAN. The EPS Bearer Identity is allocated by the MME. – 0xFF - Not Available
apnName	(optional) • APN name associated with the context id – NULL terminated by default.
ipv4Address	(optional) • IPv4 Address – 0xFFFF - Not Available

ipv4GWAddress	(optional) • IPv4 Gateway Address – 0xFFFF - Not Available
prDNSIPv4-Address	(optional) • Primary DNS IPv4 Address – 0xFFFF - Not Available
seDNSIPv4-Address	(optional) • Secondary DNS IPv4 Address – 0xFFFF - Not Available
ipv6Address	(optional) • IPv6 Address • See ipv6AddressInfo for more information
ipv6GWAddress	(optional) • IPv6 Gateway Address • See ipv6AddressInfo for more information
prDNSIPv6-Address	(optional) • Primary IPv6 DNS Address(in network byte order) • This is an 8-element array of 16-bit numbers, each of which is in big-endian format
seDNSIPv6-Address	(optional) • Secondary IPv6 DNS Address(in network byte order) • This is an 8-element array of 16-bit numbers, each of which is in big-endian format
prPCSCFIPv4-Address	(optional) • Primary PCSCF IPv4 Address
sePCSCFIPv4-Address	(optional) • Secondary PCSCF IPv4 Address
prPCSCFIPv6-Address	(optional) • Primary PCSCF IPv6 Address • This is an 8-element array of 16-bit numbers, each of which is in big-endian format
sePCSCFIPv6-Address	(optional) • Secondary PCSCF IPv6 Address • This is an 8-element array of 16-bit numbers, each of which is in big-endian format

8.730.2 Field Documentation

8.730.2.1 int8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::apnName[100]

8.730.2.2 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::bearerId

8.730.2.3 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.730.2.4 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4Address

8.730.2.5 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4GWAddress

8.730.2.6 struct ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6Address

8.730.2.7 struct ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6GWAddress

8.730.2.8 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv4Address

8.730.2.9 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv6Address[8]

8.730.2.10 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv4Address

8.730.2.11 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv6Address[8]

8.730.2.12 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv4Address

8.730.2.13 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv6Address[8]

8.730.2.14 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv4Address

8.730.2.15 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv6Address[8]

8.731 UnPackGetProfileSettingOut Struct Reference

Data Fields

- UnpackQmiProfileInfo curProfile
- uint16_t * pExtErrCode

8.731.1 Detailed Description

This structure contains the profile settings output

Parameters

curProfile	• Structure containing details of the current profile
pExtErrCode	• pointer to a 2 byte extended error code • Error code will only will be present if error code eQCWWAN_ERR_QMI_EXTENDED_INTERNAL is returned by device. • See qm_wds_ds_profile_extended_err_codes enum in qmerrno.h for received error description.

8.731.2 Field Documentation

8.731.2.1 UnpackQmiProfileInfo UnPackGetProfileSettingOut::curProfile

8.731.2.2 uint16_t* UnPackGetProfileSettingOut::pExtErrCode

8.732 unpackWdsProfileParam Union Reference

Data Fields

- LibpackProfile3GPP SlqsProfile3GPP
- LibpackProfile3GPP2 SlqsProfile3GPP2

8.732.1 Detailed Description

This union WdsProfileParam consists of Profile3GPP and Profile3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile
SlqsProfile3GP-P2	3GPP2 profile

8.732.2 Field Documentation

8.732.2.1 LibpackProfile3GPP unpackWdsProfileParam::SlqsProfile3GPP

8.732.2.2 LibpackProfile3GPP2 unpackWdsProfileParam::SlqsProfile3GPP2

8.733 voice_airTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t airTimerValue

8.733.1 Detailed Description

This structure contains information about the Air Timer.

Parameters

namID	• Index of the NAM(Number Assignment Module) to be configured. • Range 0 to 3. • Some modems support only 1 or 2 NAMs. • 0xFF,if not available.
airTimerValue	• Time in minutes. • Cumulative air time is slammed. • 0xFFFFFFFF,if not available.

8.733.2 Field Documentation

8.733.2.1 uint32_t voice_airTimer::airTimerValue

8.733.2.2 uint8_t voice_airTimer::namID

8.734 voice_allCallsAlphaIDInfo Struct Reference

Data Fields

- uint8_t callID
- voice_alphaIDInfo AlphaIDInfo

8.734.1 Detailed Description

This structure contains information for Alpha Identifier for All Calls

Parameters

callID	• Unique call identifier for the call.
AlphaIDInfo	• See voice_alphaIDInfo for more information.

8.734.2 Field Documentation

8.734.2.1 voice_alphaIDInfo voice_allCallsAlphaIDInfo::AlphaIDInfo

8.734.2.2 uint8_t voice_allCallsAlphaIDInfo::callID

8.735 voice_allCallsDiagInfo Struct Reference

Data Fields

- uint8_t callID
- voice_diagInfo DiagInfo

8.735.1 Detailed Description

This structure contains Diagnostic Information for All Calls

Parameters

callID	• Unique call identifier for the call.
DiagInfo	• See voice_diagInfo for more information.

8.735.2 Field Documentation

8.735.2.1 uint8_t voice_allCallsDiagInfo::callID

8.735.2.2 voice_diagInfo voice_allCallsDiagInfo::DiagInfo

8.736 voice_allCallsUUSInfo Struct Reference

Data Fields

- uint8_t callID
- voice_UUSInfo uusInfo

8.736.1 Detailed Description

This structure contains information for User to User Signaling Service for All Calls.

Parameters

callID	• Unique call identifier for the call.
uusInfo	• See voice_UUSInfo for more information.

8.736.2 Field Documentation

8.736.2.1 uint8_t voice_allCallsUUSInfo::callID

8.736.2.2 voice_UUSInfo voice_allCallsUUSInfo::uusInfo

8.737 voice_alphaIDInfo Struct Reference

Data Fields

- uint8_t alphaDcs
- uint8_t alphaLen
- uint8_t alphaText [255]

8.737.1 Detailed Description

This structure contains information about the Alpha Identifier.

Parameters

alphaDcs	• Alpha coding scheme – 0x01 - GSM Default_Char – 0x02 - UCS2 – 0xFF - Not Available
alphaLen	• Number of sets of the following elements: – pAlpha_text • If zero(0) then no further information exists.

alphaText[MAX_-VOICE_DESCR-IPTION_LENGT-H]	• Data encoded as per the alpha_dcs

8.737.2 Field Documentation

8.737.2.1 uint8_t voice_alphaIDInfo::alphaDcs

8.737.2.2 uint8_t voice_alphaIDInfo::alphaLen

8.737.2.3 uint8_t voice_alphaIDInfo::alphaText[255]

8.738 voice_arrAlertingPattern Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint32_t alertingPattern [20]

8.738.1 Detailed Description

This structure contains an array of Alerting Pattern.

Parameters

numInstances	• Number of callID, alertingPattern that follow. • If zero(0) then no further information exists.
callID[VOICE_-MAX_NO_OF_-CALLS]	• Array of Unique call identifier for the call.
alertingPattern[-VOICE_MAX_N-O_OF_CALLS]	• Array of Alerting pattern. – 0x00 - QMI_VOICE_ALERTING_PATTERN_1 - Pattern 1 – 0x01 - QMI_VOICE_ALERTING_PATTERN_2 - Pattern 2 – 0x02 - QMI_VOICE_ALERTING_PATTERN_3 - Pattern 3 – 0x04 - QMI_VOICE_ALERTING_PATTERN_5 - Pattern 5 – 0x05 - QMI_VOICE_ALERTING_PATTERN_6 - Pattern 6 – 0x06 - QMI_VOICE_ALERTING_PATTERN_7 - Pattern 7 – 0x07 - QMI_VOICE_ALERTING_PATTERN_8 - Pattern 8 – 0x08 - QMI_VOICE_ALERTING_PATTERN_9 - Pattern 9

8.738.2 Field Documentation

8.738.2.1 uint32_t voice_arrAlertingPattern::alertingPattern[20]

8.738.2.2 uint8_t voice_arrAlertingPattern::callID[20]

8.738.2.3 uint8_t voice_arrAlertingPattern::numInstances

8.739 voice_arrAlertingType Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint8_t AlertingType [20]

8.739.1 Detailed Description

This structure contains an array of Alerting Type.

Parameters

numInstances	• Number of callID, AlertingType that follow. • If zero(0) then no further information exists.
callID[VOICE_MAX_NO_OF_CALLS]	• Array of Unique call identifier for the call.
AlertingType[VOICE_MAX_NO_OF_CALLS]	• Array of Alerting type. – 0x00 - ALERTING_LOCAL - Local – 0x01 - ALERTING_REMOTE - Remote

8.739.2 Field Documentation

8.739.2.1 uint8_t voice_arrAlertingType::AlertingType[20]

8.739.2.2 uint8_t voice_arrAlertingType::callID[20]

8.739.2.3 uint8_t voice_arrAlertingType::numInstances

8.740 voice_arrAlphaID Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsAlphaIDInfo allCallsAlphaIDInfoArr [20]

8.740.1 Detailed Description

This structure contains an array of Alpha ID Info

Parameters

numInstances	• Number of allCallsAlphaIDInfo that follow. • If zero(0) then no further information exists.
allCallsAlphaIDInfo[VOICE_MAX_NO_OF_CALLS]	• Array of allCallsAlphaIDInfo. • See voice_allCallsAlphaIDInfo for more information.

8.740.2 Field Documentation

8.740.2.1 voice_allCallsAlphaIDInfo voice_arrAlphaID::allCallsAlphaIDInfoArr[20]

8.740.2.2 uint8_t voice_arrAlphaID::numInstances

8.741 voice_arrCalledPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo CalledPartyNum [20]

8.741.1 Detailed Description

This structure contains an array of Called Party Numbers consisting of information of all the numbers which have been called from the device.

Parameters

numInstances	• Number of calledPartyNum that follow. • If zero(0) then no further information exists.
CalledPartyNum[VOICE_MAX_NO_OF_CALLS]	• Array of CalledPartyNum. • See voice_peerNumberInfo for more information.

8.741.2 Field Documentation

8.741.2.1 voice_peerNumberInfo voice_arrCalledPartyNum::CalledPartyNum[20]

8.741.2.2 uint8_t voice_arrCalledPartyNum::numInstances

8.742 voice_arrCallEndReason Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t callEndReason [20]

8.742.1 Detailed Description

This structure contains an array of Call End Reasons.

Parameters

numInstances	• Number of callID, callEndReason that follow. • If zero(0) then no further information exists.
callID[VOICE_MAX_NO_OF_CALLS]	• Array of Unique call identifier for the call.
callEndReason[VOICE_MAX_NO_OF_CALLS]	• Array of Call End Reason . • See Table9 qaGobiApiTableVoiceCallEndReasons.h for a list of valid voice-related call end reasons

8.742.2 Field Documentation

8.742.2.1 uint16_t voice_arrCallEndReason::callEndReason[20]

8.742.2.2 uint8_t voice_arrCallEndReason::callID[20]

8.742.2.3 uint8_t voice_arrCallEndReason::numInstances

8.743 voice_arrCallInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallInformation getAllCallInfo [20]

8.743.1 Detailed Description

This structure contains an array of Call Info

Parameters

numInstances	• Number of getAllCallInfo that follow. • If zero(0) then no further information exists.
getAllCallInfo[VOICE_MAX_NO_OF_CALLS]	• Array of CallInfo. • See voice_getAllCallInformation for more information.

8.743.2 Field Documentation

8.743.2.1 voice_getAllCallInformation voice_arrCallInfo::getAllCallInfo[20]

8.743.2.2 uint8_t voice_arrCallInfo::numInstances

8.744 voice_arrConnectPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo ConnectedPartyNum [20]

8.744.1 Detailed Description

This structure contains an array of Connected Party Numbers consisting of information regarding all the devices connected.

Parameters

numInstances	• Number of ConnectedPartyNum that follow. • If zero(0) then no further information exists.
ConnectedPartyNum[VOICE_MAX_NO_OF_CALLS]	• Array of ConnectedPartyNum. • See voice_peerNumberInfo for more information.

8.744.2 Field Documentation

8.744.2.1 voice_peerNumberInfo voice_arrConnectPartyNum::ConnectedPartyNum[20]

8.744.2.2 uint8_t voice_arrConnectPartyNum::numInstances

8.745 voice_arrDiagInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsDiagInfo DiagInfo [20]

8.745.1 Detailed Description

This structure contains an array of Diagnostic Information.

Parameters

numInstances	• Number of DiagInfo that follow. • If zero(0) then no further information exists.
DiagInfo[VOICE_MAX_NO_OF_CALLS]	• Array of DiagInfo. • See voice_allCallsDiagInfo for more information.

8.745.2 Field Documentation

8.745.2.1 voice_allCallsDiagInfo voice_arrDiagInfo::DiagInfo[20]

8.745.2.2 uint8_t voice_arrDiagInfo::numInstances

8.746 voice_arrRedirPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo RedirPartyNum [20]

8.746.1 Detailed Description

This structure contains an array of Redirecting Party Numbers consisting of information of all the numbers which have been redirected from the device.

Parameters

numInstances	• Number of redirPartyNum that follow. • If zero(0) then no further information exists.
RedirPartyNum[-VOICE_MAX_NO_OF_CALLS]	• Array of RedirPartyNum. • See voice_peerNumberInfo for more information.

8.746.2 Field Documentation

8.746.2.1 uint8_t voice_arrRedirPartyNum::numInstances

8.746.2.2 voice_peerNumberInfo voice_arrRedirPartyNum::RedirPartyNum[20]

8.747 voice_arrRemotePartyName Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyName GetAllCallRmtPtyName [20]

8.747.1 Detailed Description

This structure contains an array of Remote Party Names

Parameters

numInstances	• Number of remotePartyName that follow. • If zero(0) then no further information exists.

GetAllCallRmtPtyName[MAX_NO_OF_CALLS]	• Array of remotePartyName. • See voice_getAllCallRmtPtyName for more information.

8.747.2 Field Documentation

8.747.2.1 voice_getAllCallRmtPtyName voice_arrRemotePartyName::GetAllCallRmtPtyName[20]

8.747.2.2 uint8_t voice_arrRemotePartyName::numInstances

8.748 voice_arrRemotePartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyNum RmtPtyNum [20]

8.748.1 Detailed Description

This structure contains an array of Remote Party Numbers

Parameters

numInstances	• Number of remotePartyNum that follow. • If zero(0) then no further information exists.
RmtPtyNum[VOICE_MAX_NO_OF_CALLS]	• Array of remotePartyNum. • See voice_getAllCallRmtPtyNum for more information.

8.748.2 Field Documentation

8.748.2.1 uint8_t voice_arrRemotePartyNum::numInstances

8.748.2.2 voice_getAllCallRmtPtyNum voice_arrRemotePartyNum::RmtPtyNum[20]

8.749 voice_arrSvcOption Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t srvOption [20]

8.749.1 Detailed Description

This structure contains array an of Servicing option.

Parameters

numInstances	• Number of callID, srvOption that follow. • If zero(0) then no further information exists.
callID[VOICE_MAX_NO_OF_CALLS]	• Array of Unique call identifier for the call.
srvOption[VOICE_MAX_NO_OF_CALLS]	• Array of Service option. • See Table9 qaGobiApiTableServiceOptions.h for standard service option number assignments.

8.749.2 Field Documentation

8.749.2.1 uint8_t voice_arrSvcOption::callID[20]

8.749.2.2 uint8_t voice_arrSvcOption::numInstances

8.749.2.3 uint16_t voice_arrSvcOption::srvOption[20]

8.750 voice_arrUUSInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsUUSInfo AllCallsUUSInfo [20]

8.750.1 Detailed Description

This structure contains an array of User to User Signaling Service Information

Parameters

numInstances	• Number of allCallsUUSInfo that follow. • If zero(0) then no further information exists.
AllCallsUUSInfo[VOICE_MAX_NO_OF_CALLS]	• Array of allCallsUUSInfo. • See voice_allCallsUUSInfo for more information.

8.750.2 Field Documentation

8.750.2.1 voice_allCallsUUSInfo voice_arrUUSInfo::AllCallsUUSInfo[20]

8.750.2.2 uint8_t voice_arrUUSInfo::numInstances

8.751 voice_burstDTMFInfo Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t digitCnt
- uint8_t pDigitBuff [255]

8.751.1 Detailed Description

This structure contains Voice Burst DTMF Information

Parameters

pCallID	• Call ID associated with call on which the DTMF information has to be sent. A burst DTMF request is sent to the current active/alerting call when pCallId is set to 0xFF. • This is IN/OUT parameter, value passed by user will be packed in request and value received from the device would be returned to the user. • If the call ID value received is 0, no value has been returned by the device
digitCnt	• Length of DTMF digit buffer which follows
pDigitBuff[MAX_VOICE_DESCRIPTION_LENGTH]	• DTMF digit buffer in ASCII, NULL terminated

8.751.2 Field Documentation

8.751.2.1 uint8_t voice_burstDTMFInfo::digitCnt

8.751.2.2 uint8_t* voice_burstDTMFInfo::pCallID

8.751.2.3 uint8_t voice_burstDTMFInfo::pDigitBuff[255]

8.752 voice_calledPartySubAdd Struct Reference

Data Fields

- uint8_t extBit
- uint8_t subAddrType
- uint8_t oddEvenInd
- uint8_t subAddrLen
- uint8_t subAddr [255]

8.752.1 Detailed Description

This structure contains information about the Called Sub Party Addresses.

Parameters

extBit	• Extension bit.

subAddrType	• Subaddress type. – 0x00 - NSAP – 0x01 - USER
oddEvenInd	• Even/odd indicator. – 0x00 - Even number of address signals – 0x01 - Odd number of address signals
subAddrLen	• Number of sets of the following elements: – SubAddress
subAddr[MAX_VOICE_DESCRIPTION_LENGTH]	• Array of the SubAddress in BCD number format.

8.752.2 Field Documentation

8.752.2.1 uint8_t voice_calledPartySubAdd::extBit

8.752.2.2 uint8_t voice_calledPartySubAdd::oddEvenInd

8.752.2.3 uint8_t voice_calledPartySubAdd::subAddr[255]

8.752.2.4 uint8_t voice_calledPartySubAdd::subAddrLen

8.752.2.5 uint8_t voice_calledPartySubAdd::subAddrType

8.753 voice_callFwdTypeAndPlan Struct Reference

Data Fields

- uint8_t numberType
- uint8_t numberPlan

8.753.1 Detailed Description

This structure contains Supplementary Service request parameters related to different features and their activation, deactivation, registration and erasure (applicable only for 3GPP)

Parameters

numberType	• Call forwarding number type – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION Reserved extension
numberPlan	• Call forwarding number plan – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION Reserved extension

8.753.2 Field Documentation

8.753.2.1 uint8_t voice_callFwdTypeAndPlan::numberPlan

8.753.2.2 uint8_t voice_callFwdTypeAndPlan::numberType

8.754 voice_callFWExtInfo Struct Reference

Data Fields

- uint8_t SvcStatus
- uint8_t SvcClass
- uint8_t noReplyTimer
- uint8_t PI
- uint8_t SI
- uint8_t numType
- uint8_t numPlan
- uint8_t numLen
- uint8_t number [255]

8.754.1 Detailed Description

This structure contains information for Get Call Forwarding Extended Information.

Parameters

SvcStatus	• Service status. Values: – 0x00 - SERVICE_STATUS_INACTIVE - Inactive – 0x01 - SERVICE_STATUS_ACTIVE - Active
SvcClass	• Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.
noReplyTimer	• No reply timer value in seconds • A value of 0 indicates that no_reply_timer is ignored.
PI	• Presentation indicator; refer to [S1, Table 2.7.4.4-1] for valid values.
SI	• Number screening indicator. • Values: – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
numType	• Number type. • Values: – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
numPlan	• Number plan. • Values: – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension

numLen	• Provides the length of number which follow.
number[255]	• number of numLen length, NULL terminated.

8.754.2 Field Documentation

8.754.2.1 uint8_t voice_callFWExtInfo::noReplyTimer

8.754.2.2 uint8_t voice_callFWExtInfo::number[255]

8.754.2.3 uint8_t voice_callFWExtInfo::numLen

8.754.2.4 uint8_t voice_callFWExtInfo::numPlan

8.754.2.5 uint8_t voice_callFWExtInfo::numType

8.754.2.6 uint8_t voice_callFWExtInfo::PI

8.754.2.7 uint8_t voice_callFWExtInfo::SI

8.754.2.8 uint8_t voice_callFWExtInfo::SvcClass

8.754.2.9 uint8_t voice_callFWExtInfo::SvcStatus

8.755 voice_callFWInfo Struct Reference

Data Fields

- uint8_t SvcStatus
- uint8_t SvcClass
- uint8_t numLen
- uint8_t number [255]
- uint8_t noReplyTimer

8.755.1 Detailed Description

This structure contains information for Get Call Forwarding Information.

Parameters

SvcStatus	• Service status. Values: – 0x00 - SERVICE_STATUS_INACTIVE - Inactive – 0x01 - SERVICE_STATUS_ACTIVE - Active
SvcClass	• Service Class is a combination (sum) of information class constants • See qaGobiApiTableSupServiceInfoClasses.h for service classes.

numLen	• Provides the length of number which follow.
number[255]	• number of numLen length, NULL terminated.
noReplyTimer	• No reply timer value in seconds • A value of 0 indicates that noReplyTimer is ignored.

8.755.2 Field Documentation

8.755.2.1 uint8_t voice_callFWInfo::noReplyTimer

8.755.2.2 uint8_t voice_callFWInfo::number[255]

8.755.2.3 uint8_t voice_callFWInfo::numLen

8.755.2.4 uint8_t voice_callFWInfo::SvcClass

8.755.2.5 uint8_t voice_callFWInfo::SvcStatus

8.756 voice_callInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t callState
- uint8_t callType
- uint8_t direction
- uint8_t mode

8.756.1 Detailed Description

This structure contains Information about call state changes. For example, when an incoming call is received, this structure is populated and indicate the incoming call information. When this incoming call is answered, the call status changes from INCOMING to CONVERSATION, which means a change in the call information and this structure is populated again with the changes and notified to/retrived by the user.

Parameters

callID	• Call identifier for the call queried for information. • If zero(0) then invalid.
callState	• Call state. – 0x01 - CALL_STATE_ORIGINATION - Origination – 0x02 - CALL_STATE_INCOMING - Incoming – 0x03 - CALL_STATE_CONVERSATION - Conversation – 0x04 - CALL_STATE_CC_IN_PROGRESS - Call is originating but waiting for call control to complete – 0x05 - CALL_STATE_ALERTING - Alerting – 0x06 - CALL_STATE_HOLD - Hold – 0x07 - CALL_STATE_WAITING - Waiting – 0x08 - CALL_STATE_DISCONNECTING - Disconnecting – 0x09 - CALL_STATE_END - End – 0x0A - CALL_STATE_SETUP - MT call is in Setup state in 3GPP – 0xFF - Not Available
callType	• Call type. – 0x00 - CALL_TYPE_VOICE - Voice – 0x02 - CALL_TYPE_VOICE_IP - Voice over IP – 0x06 - CALL_TYPE_OTAPA - OTAPA – 0x07 - CALL_TYPE_STD_OTASP - Standard OTASP – 0x08 - CALL_TYPE_NON_STD_OTASP - Nonstandard OTASP – 0x09 - CALL_TYPE_EMERGENCY - Emergency – 0xFF - Not Available
direction	• Direction. – 0x01 - CALL_DIRECTION_MO - MO call – 0x02 - CALL_DIRECTION_MT - MT call – 0xFF - Not Available
mode	• Mode. • If the mode field is "0x01 - CDMA",the optional Service Option, Voice Privacy, and OTASP Status (only for OTASP calls) TLVs are included in the response. – 0x01 - CALL_MODE_CDMA - CDMA – 0x02 - CALL_MODE_GSM - GSM – 0x03 - CALL_MODE_UMTS - UMTS – 0x04 - CALL_MODE_LTE - LTE – 0x05 - CALL_MODE_TDS - TD-SCDMA – 0xFF - Not Available

8.756.2 Field Documentation

8.756.2.1 uint8_t voice_callInfo::callID

8.756.2.2 uint8_t voice_callInfo::callState

8.756.2.3 uint8_t voice_callInfo::callType

8.756.2.4 uint8_t voice_callInfo::direction

8.756.2.5 uint8_t voice_callInfo::mode

8.757 voice_ccSUPSType Struct Reference

Data Fields

- uint8_t svcType
- uint8_t reason

8.757.1 Detailed Description

This structure contains information about the Call Control Supplementary Service Types

Parameters

svcType	• Service type. – 0x01 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_ACTIVATE - Activate – 0x02 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_DEACTIVATE - Deactivate – 0x03 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_REGISTER - Register – 0x04 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_ERASE - Erase – 0x05 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_INTERROGATE - Interrogate – 0x06 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_REGISTER_PASSWORD - Register password – 0x07 - VOICE_CC_SUPS_RESULT_SERVICE_TYPE_USSD - USSD – 0xFF - Not Available
reason	• Call control supplementary service result reason • Values: – See qaGobiApiTableCallControlReturnReasons.h for return reasons.

8.757.2 Field Documentation

8.757.2.1 uint8_t voice_ccSUPSType::reason

8.757.2.2 uint8_t voice_ccSUPSType::svcType

8.758 voice_CLIPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.758.1 Detailed Description

This structure contains information about the Calling Line Identification Presentation (CLIP) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.758.2 Field Documentation

8.758.2.1 uint8_t voice_CLIPResp::ActiveStatus

8.758.2.2 uint8_t voice_CLIPResp::ProvisionStatus

8.759 voice_CLIRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.759.1 Detailed Description

This structure contains information about the Calling Line Identification Restriction (CLIR) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available

ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED_PERMANENT - Permanently provisioned – 0x02 - PROVISION_STATUS_PRESENTATION_RESTRICTED - Restricted presentation – 0x03 - PROVISION_STATUS_PRESENTATION_ALLOWED - Allowed presentation – 0xFF - Not Available

8.759.2 Field Documentation

8.759.2.1 uint8_t voice_CLIRResp::ActiveStatus

8.759.2.2 uint8_t voice_CLIRResp::ProvisionStatus

8.760 voice_CNAPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.760.1 Detailed Description

This structure contains information about the Calling Name Presentation (CNAP) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.760.2 Field Documentation

8.760.2.1 uint8_t voice_CNAPResp::ActiveStatus

8.760.2.2 uint8_t voice_CNAPResp::ProvisionStatus

8.761 voice_COLPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.761.1 Detailed Description

This structure contains information about the Connected Line Identification Presentation (COLP) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.761.2 Field Documentation

8.761.2.1 uint8_t voice_COLPResp::ActiveStatus

8.761.2.2 uint8_t voice_COLPResp::ProvisionStatus

8.762 voice_COLRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.762.1 Detailed Description

This structure contains information about the Connected Line Identification Restriction (COLR) supplementary service responses.

Parameters

ActiveStatus	• Active status. • Values: – 0x00 - ACTIVE_STATUS_INACTIVE - Inactive – 0x01 - ACTIVE_STATUS_ACTIVE - Active – 0xFF - Not Available
ProvisionStatus	• Provisioned status. • Values: – 0x00 - PROVISION_STATUS_NOT_PROVISIONED - Not provisioned – 0x01 - PROVISION_STATUS_PROVISIONED - Provisioned – 0xFF - Not Available

8.762.2 Field Documentation

8.762.2.1 uint8_t voice_COLRResp::ActiveStatus

8.762.2.2 uint8_t voice_COLRResp::ProvisionStatus

8.763 voice_connectNumInfo Struct Reference

Data Fields

- uint8_t numPresInd
- uint8_t screeningInd
- uint8_t numType
- uint8_t numPlan
- uint8_t callerIDLen
- uint8_t callerID [81]

8.763.1 Detailed Description

This structure contains information about the numbers connected to a device. It contains information such as number type, eg International or Local.

Parameters

numPresInd	• Presentation indicator – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available

screeningInd	• Screening indicator. – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network – 0xFF - Not Available
numType	• Number type. – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension – 0xFF - Not Available
numPlan	• Number plan. – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension – 0xFF - Not Available
callerIDLen	• Provides the length of caller ID which follow. • If zero(0) then no further information exists.
callerID[MAX_VOICE_CALL_NO_LEN]	• callerID of numLen length, NULL terminated.

8.763.2 Field Documentation

8.763.2.1 uint8_t voice_connectNumInfo::callerID[81]

8.763.2.2 uint8_t voice_connectNumInfo::callerIDLen

8.763.2.3 uint8_t voice_connectNumInfo::numPlan

8.763.2.4 uint8_t voice_connectNumInfo::numPresInd

8.763.2.5 uint8_t voice_connectNumInfo::numType

8.763.2.6 uint8_t voice_connectNumInfo::screeningInd

8.764 voice_CUGInfo Struct Reference

Data Fields

- uint16_t CUGIndex
- uint8_t SuppPrefCUG
- uint8_t SuppOA

8.764.1 Detailed Description

This structure contains Closed User Group Information

Parameters

CUGIndex	• Range 0x00... 0x7FFF
SuppPrefCUG	• Suppress preferential CUG – 0x00 - FALSE – 0x01 - TRUE
SuppOA	• Suppress OA subscription option – 0x00 - FALSE – 0x01 - TRUE

8.764.2 Field Documentation

8.764.2.1 uint16_t voice_CUGInfo::CUGIndex

8.764.2.2 uint8_t voice_CUGInfo::SuppOA

8.764.2.3 uint8_t voice_CUGInfo::SuppPrefCUG

8.765 voice_curAMRConfig Struct Reference

Data Fields

- uint8_t gsmAmrStat
- uint8_t wcdmaAmrStat

8.765.1 Detailed Description

This structure contains the Current Adaptive Multi Rate Configuration Information.

Parameters

gsmAmrStat	• GSM AMR Status – 0x00 - Disable – 0x01 - Enable – 0xFF - Not Available
wcdmaAmrStat	• WCDMA AMR Status • One or a combination of the following bitmask values: – Bit 0 - AMR codec advertised is not supported – Bit 1 - Controls WCDMA AMR wideband – Bit 2 - Controls GSM half rate AMR – Bit 3 - Controls GSM AMR wideband – Bit 4 - Controls GSM AMR narrowband • 0xFF, if not available

8.765.2 Field Documentation

8.765.2.1 uint8_t voice_curAMRConfig::gsmAmrStat

8.765.2.2 uint8_t voice_curAMRConfig::wcdmaAmrStat

8.766 voice_diagInfo Struct Reference

Data Fields

- uint8_t diagInfoLen
- uint8_t diagnosticInfo [255]

8.766.1 Detailed Description

This structure contains Diagnostic Information

Parameters

diagInfoLen	• Provides the length of information which follow. • If zero(0) then no further information exists.
diagnosticInfo[MAX_VOICE_DESCRIPTION_LENGTH]	• Diagnostic information.

8.766.2 Field Documentation

8.766.2.1 uint8_t voice_diagInfo::diagInfoLen

8.766.2.2 uint8_t voice_diagInfo::diagnosticInfo[255]

8.767 voice_DTMFLengths Struct Reference

Data Fields

- uint8_t DTMFPulseWidth
- uint8_t DTMFInterdigitInterval

8.767.1 Detailed Description

This structure contains Voice Burst DTMF pulse length information

Parameters

DTMFPulse-Width	• DTMF pulse width. Values: – 0x00 - DTMF_ONLENGTH_95MS - 95 ms – 0x01 - DTMF_ONLENGTH_150MS - 150 ms – 0x02 - DTMF_ONLENGTH_200MS - 200 ms – 0x03 - DTMF_ONLENGTH_250MS - 250 ms – 0x04 - DTMF_ONLENGTH_300MS - 300 ms – 0x05 - DTMF_ONLENGTH_350MS - 350 ms – 0x06 - DTMF_ONLENGTH_SMS SMS Tx special pulse width
DTMFInterdigit-Interval	• DTMF interdigit interval Values: – 0x00 - DTMF_OFFLENGTH_60MS - 60 ms – 0x01 - DTMF_OFFLENGTH_100MS - 100 ms – 0x02 - DTMF_OFFLENGTH_150MS - 150 ms – 0x03 - DTMF_OFFLENGTH_200MS - 200 ms

8.767.2 Field Documentation

8.767.2.1 uint8_t voice_DTMFLengths::DTMFInterdigitInterval

8.767.2.2 uint8_t voice_DTMFLengths::DTMFPulseWidth

8.768 voice_getAllCallInformation Struct Reference

Data Fields

- voice_callInfo CallInfo
- uint8_t isEmpty
- uint8_t ALS

8.768.1 Detailed Description

This structure contains information related to call state change.

Parameters

CallInfo	• See voice_callInfo for more information.
isEmpty	• Multiparty indicator. – 0x00 - False – 0x01 - True
ALS	• Alternate Line Service line indicator. • Feature for supporting two different phone numbers on the same mobile device. – 0x00 - ALS_LINE1 - Line 1 (default) – 0x01 - ALS_LINE2 - Line 2

8.768.2 Field Documentation

8.768.2.1 uint8_t voice_getAllCallInformation::ALS

8.768.2.2 voice_callInfo voice_getAllCallInformation::CallInfo

8.768.2.3 uint8_t voice_getAllCallInformation::isEmpty

8.769 voice_getAllCallRmtPtyName Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyName RemotePartyName

8.769.1 Detailed Description

This structure contains information for All Call Remote Party Names

Parameters

callID	• Unique call identifier for the call.
RemoteParty-Name	• See voice_remotePartyName for more information.

8.769.2 Field Documentation

8.769.2.1 uint8_t voice_getAllCallRmtPtyName::callID

8.769.2.2 voice_remotePartyName voice_getAllCallRmtPtyName::RemotePartyName

8.770 voice_getAllCallRmtPtyNum Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyNum RemotePartyNum

8.770.1 Detailed Description

This structure contains information for All Call Remote Party Numbers

Parameters

callID	• Unique call identifier for the call.
RemoteParty-Num	• See voice_remotePartyNum for more information.

8.770.2 Field Documentation

8.770.2.1 uint8_t voice_getAllCallRmtPtyNum::callID

8.770.2.2 voice_remotePartyNum voice_getAllCallRmtPtyNum::RemotePartyNum

8.771 voice_getCallFWExtInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWExtInfo CallFWExtInfo [20]

8.771.1 Detailed Description

This structure contains an array of Call Forwarded Extended Information.

Parameters

numInstances	• Number of callFWExtInfo that follow. • If zero(0) then no further information exists.
CallFWExtInfo[-VOICE_MAX_N-O_OF_CALLS]	• Array of CallFWExtInfo. – See voice_callFWExtInfo for more information.

8.771.2 Field Documentation

8.771.2.1 voice_callFWExtInfo voice_getCallFWExtInfo::CallFWExtInfo[20]

8.771.2.2 uint8_t voice_getCallFWExtInfo::numInstances

8.772 voice_getCallFWInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWInfo CallFWInfo [20]

8.772.1 Detailed Description

This structure contains an array of Call Forwarded Information.

Parameters

numInstances	• Number of callFWInfo that follow. • If zero(0) then no further information exists.
CallFWInfo[VOICE_MAX_NO_OF_CALLS]	• Array of callFWInfo. – See voice_callFWInfo for more information.

8.772.2 Field Documentation

8.772.2.1 voice_callFWInfo voice_getCallFWInfo::CallFWInfo[20]

8.772.2.2 uint8_t voice_getCallFWInfo::numInstances

8.773 voice_peerNumberInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t numPI
- uint8_t numSI
- uint8_t numType
- uint8_t numPlan
- uint8_t numLen
- uint8_t number [81]

8.773.1 Detailed Description

This structure contains information for Connected Peer Numbers.

Parameters

callID	• Unique call identifier for the call.
numPI	• Number presentation indicator. – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available
numSI	• Number screening indicator. – 0x00 - QMI_VOICE_SI_USER_PROVIDED_NOT_SCREENED - Provided user is not screened – 0x01 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_PASSED - Provided user passed verification – 0x02 - QMI_VOICE_SI_USER_PROVIDED_VERIFIED_FAILED - Provided user failed verification – 0x03 - QMI_VOICE_SI_NETWORK_PROVIDED - Provided network
numType	• Number type. – 0x00 - QMI_VOICE_NUM_TYPE_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_TYPE_INTERNATIONAL - International – 0x02 - QMI_VOICE_NUM_TYPE_NATIONAL - National – 0x03 - QMI_VOICE_NUM_TYPE_NETWORK_SPECIFIC - Network-specific – 0x04 - QMI_VOICE_NUM_TYPE_SUBSCRIBER - Subscriber – 0x05 - QMI_VOICE_NUM_TYPE_RESERVED - Reserved – 0x06 - QMI_VOICE_NUM_TYPE_ABBREVIATED - Abbreviated – 0x07 - QMI_VOICE_NUM_TYPE_RESERVED_EXTENSION - Reserved extension
numPlan	• Number plan. – 0x00 - QMI_VOICE_NUM_PLAN_UNKNOWN - Unknown – 0x01 - QMI_VOICE_NUM_PLAN_ISDN - ISDN – 0x03 - QMI_VOICE_NUM_PLAN_DATA - Data – 0x04 - QMI_VOICE_NUM_PLAN_TELEX - Telex – 0x08 - QMI_VOICE_NUM_PLAN_NATIONAL - National – 0x09 - QMI_VOICE_NUM_PLAN_PRIVATE - Private – 0x0B - QMI_VOICE_NUM_PLAN_RESERVED_CTS - Reserved cordless telephony system – 0x0F - QMI_VOICE_NUM_PLAN_RESERVED_EXTENSION - Reserved extension
numLen	• Provides the length of number which follow.
number[MAX_VOICE_CALL_NO_LEN]	• number of numLen length, NULL terminated.

8.773.2 Field Documentation

8.773.2.1 uint8_t voice_peerNumberInfo::callID

8.773.2.2 uint8_t voice_peerNumberInfo::number[81]

8.773.2.3 uint8_t voice_peerNumberInfo::numLen

8.773.2.4 uint8_t voice_peerNumberInfo::numPI

8.773.2.5 uint8_t voice_peerNumberInfo::numPlan

8.773.2.6 uint8_t voice_peerNumberInfo::numSI

8.773.2.7 uint8_t voice_peerNumberInfo::numType

8.774 voice_prefVoiceSO Struct Reference

Data Fields

- uint8_t namID
- uint8_t evrcCapability
- uint16_t homePageVoiceSO
- uint16_t homeOrigVoiceSO
- uint16_t roamOrigVoiceSO

8.774.1 Detailed Description

This structure contains information about the Preferred Voice Service Options.

Parameters

namID	• Index of the NAM(Number Assignment Module) to be configured. • Range 0 to 3. • Some modems support only 1 or 2 NAMs. • 0xFF,if not available.
evrcCapability	• EVRC capability. • Values: – 0x00 - Disable – 0x01 - Enable – 0xFF - Not Available

homePageVoiceSO	• Home page voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARR0W_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available
homeOrigVoiceSO	• Home origination voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARR0W_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available
roamOrigVoiceSO	• Roaming origination voice SO; most preferred CDMA SO to be requested from the network when receiving an incoming (MT) voice call within the home network. • Values: – 0x0000 - VOICE_SO_WILD - Any service option – 0x0001 - VOICE_SO_IS_96A - IS-96A – 0x0003 - VOICE_SO_EVRC - EVRC – 0x0011 - VOICE_SO_13K_IS733 - 13K_IS733 – 0x0038 - VOICE_SO_SELECTABLE_MODE_VOCODER - Selectable mode vocoder – 0x0044 - VOICE_SO_4GV_NARR0W_BAND - 4GV narrowband – 0x0046 - VOICE_SO_4GV_WIDE_BAND - 4GV wideband – 0x8000 - VOICE_SO_13K - 13K – 0x8001 - VOICE_SO_IS_96 - IS-96 – 0x8023 - VOICE_SO_WVRC - WVRC – 0xFFFF - Not Available

8.774.2 Field Documentation

8.774.2.1 uint8_t voice_prefVoiceSO::evrcCapability

8.774.2.2 uint16_t voice_prefVoiceSO::homeOrigVoiceSO

8.774.2.3 uint16_t voice_prefVoiceSO::homePageVoiceSO

8.774.2.4 uint8_t voice_prefVoiceSO::namID

8.774.2.5 uint16_t voice_prefVoiceSO::roamOrigVoiceSO

8.775 voice_remotePartyName Struct Reference

Data Fields

- uint8_t namePI
- uint8_t codingScheme
- uint8_t nameLen
- uint8_t callerName [255]

8.775.1 Detailed Description

This structure contains information about the names that are dialed from the device or from which a call is received on the device.

Parameters

namePI	• Name presentation indicator. – 0x00 - PRESENTATION_NAME_PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_NAME_PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NAME_UNAVAILABLE - Unavailable presentation – 0x03 - PRESENTATION_NAME_NAME_PRESENTATION_RESTRICTED - Restricted name presentation – 0xFF - Not Available
codingScheme	• Refer to Table10 qaGobiApiTableCodingScheme.h for coding schemes • 0xFF - Not Available
nameLen	• Provides the length of name which follow. • If zero(0) then no further information exists.
callerName[MAX_VOICE_DESCRIPTION_LENGTH]	• Name in ASCII, NULL ending.

8.775.2 Field Documentation

8.775.2.1 uint8_t voice_remotePartyName::callerName[255]

8.775.2.2 uint8_t voice_remotePartyName::codingScheme

8.775.2.3 uint8_t voice_remotePartyName::nameLen

8.775.2.4 uint8_t voice_remotePartyName::namePI

8.776 voice_remotePartyNum Struct Reference

Data Fields

- uint8_t presentationInd
- uint8_t numLen
- uint8_t remPartyNumber [81]

8.776.1 Detailed Description

This structure contains information about the numbers that are dialed from the device or from which a call is received on the device.

Parameters

presentationInd	• Presentation indicator. – 0x00 - PRESENTATION_ALLOWED - Allowed presentation – 0x01 - PRESENTATION_RESTRICTED - Restricted presentation – 0x02 - PRESENTATION_NUM_UNAVAILABLE - Unavailable presentation – 0x04 - PRESENTATION_PAYPHONE - Payphone presentation (GSM/UMTS specific) – 0xFF - Not Available
numLen	• Provides the length of number which follow. • If zero(0) then no further information exists.
remParty-Number[MAX_-CALL_NO_LEN]	• Array of numbers in ASCII, NULL ending.

8.776.2 Field Documentation

8.776.2.1 uint8_t voice_remotePartyNum::numLen

8.776.2.2 uint8_t voice_remotePartyNum::presentationInd

8.776.2.3 uint8_t voice_remotePartyNum::remPartyNumber[81]

8.777 voice_roamTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t roamTimerValue

8.777.1 Detailed Description

This structure contains information about the Roam Timer.

Parameters

namID	• Index of the NAM(Number Assignment Module) to be configured. • Range 0 to 3. • Some modems support only 1 or 2 NAMs. • 0xFF,if not available.
roamTimerValue	• Time in minutes. • Cumulative air time is slammed. • 0xFFFFFFFF,if not available.

8.777.2 Field Documentation

8.777.2.1 uint8_t voice_roamTimer::namID

8.777.2.2 uint32_t voice_roamTimer::roamTimerValue

8.778 voice_USSInfo Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.778.1 Detailed Description

This structure contains USS Information

Parameters

ussDCS	• 1 - ASCII coding scheme • 2 - 8-BIT coding scheme • 3 - UCS2
ussLen	• Range 1 to 182
ussData	• Data encoded as per the DCS

8.778.2 Field Documentation

8.778.2.1 uint8_t voice_USSInfo::ussData[182]

8.778.2.2 uint8_t voice_USSInfo::ussDCS

8.778.2.3 uint8_t voice_USSInfo::ussLen

8.779 voice_UUSInfo Struct Reference

Data Fields

- uint8_t UUSType
- uint8_t UUSDcs
- uint8_t UUSDatalen
- uint8_t UUSData [255]

8.779.1 Detailed Description

This structure contains User to User Signaling Service Information.

Parameters

UUSType	• UUS type values are: – 0x00 - UUS_DATA – 0x01 - UUS_TYPE1_IMPLICIT – 0x02 - UUS_TYPE1_REQUIRED – 0x03 - UUS_TYPE1_NOT_REQUIRED – 0x04 - UUS_TYPE2_REQUIRED – 0x05 - UUS_TYPE2_NOT_REQUIRED – 0x06 - UUS_TYPE3_REQUIRED – 0x07 - UUS_TYPE3_NOT_REQUIRED – 0xFF - Not Available
UUSDcs	• UUS data coding scheme values are: – 0x01 - UUS_DCS_USP – 0x02 - UUS_DCS_OHLP – 0x03 - UUS_DCS_X244 – 0x04 - UUS_DCS_SMCF – 0x05 - UUS_DCS_IA5 – 0x06 - UUS_DCS_RV12RD – 0x07 - UUS_DCS_Q931UNCCM – 0xFF - Not Available
UUSDatalen	• Number of sets of the following elements. – UUSData • If zero(0) then no further information exists.
UUSData[MAX_VOICE_DESCRIPTION_LENGTH]	• UUS data encoded as per coding scheme

8.779.2 Field Documentation

8.779.2.1 uint8_t voice_UUSInfo::UUSData[255]

8.779.2.2 uint8_t voice_UUSInfo::UUSDatalen

8.779.2.3 uint8_t voice_UUSInfo::UUSDcs

8.779.2.4 uint8_t voice_UUSInfo::UUSType

8.780 wds_currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.780.1 Detailed Description

Network information structure

Parameters

NetworkType	• Values: – 0 - 3GPP – 1 - 3GPP2
RATMask	• Radio Access Technology (RAT) mask to indicate the type of technology. • Values: – 0 - Don't Care – 0x8000 - NULL Bearer • CDMA RAT mask values: – 0x01 - CDMA_1x – 0x02 - EVDO_REV0 – 0x05 - HRPD – 0x0B - EHRPD • UMTS RAT mask values: – 0x03 - GPRS – 0x04 - WCDMA – 0x06 - EDGE – 0x07 - HSDPA and WCDMA – 0x08 - HSUPA and WCDMA – 0x08 - HSDPA and HSUPA – 0x0A - LTE – 0x0C - HSDPA+ and WCDMA – 0x0D - HSDPA+ and HSUPA – 0x0E - DC_HSDPA+ and WCDMA – 0x0F - DC_HSDPA+ and HSUPA

SOMask	• Service Option (SO) mask to indicate the service option or type of application. • Values: – 0 - Don't Care • CDMA 1x SO mask values: – 0x01 - CDMA_1X_IS95 – 0x02 - CDMA_1X_IS2000 – 0x04 - CDMA_1X_IS2000_REL_A • CDMA EV-DO Rev 0 SO mask values: – 0x01 - DPA • CDMA EV-DO Rev A SO mask values: – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD • CDMA EV-DO Rev B SO mask values: – 0x01 - DPA – 0x02 - MFPA – 0x04 - EMPA – 0x08 - EMPA_EHRPD – 0x10 - MMPA – 0x20 - MMPA_EHRPD

8.780.2 Field Documentation

8.780.2.1 uint8_t wds_currNetworkInfo::NetworkType

8.780.2.2 uint32_t wds_currNetworkInfo::RATMask

8.780.2.3 uint32_t wds_currNetworkInfo::SOMask

8.781 wds_DataULongLongTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t ullData

8.781.1 Detailed Description

Parameters

TlvPresent	a flag indicates whether TLV presented or not
ulData	unsigend long long data in 64bit

8.781.2 Field Documentation

8.781.2.1 uint8_t wds_DataULongLongTlv::TlvPresent

8.781.2.2 uint64_t wds_DataULongLongTlv::ullData

8.782 wds_DataULongTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t ulData

8.782.1 Detailed Description

Parameters

TlvPresent	a flag indicates whether TLV presented or not
ulData	unsigend long data in 32bit

8.782.2 Field Documentation

8.782.2.1 uint8_t wds_DataULongTlv::TlvPresent

8.782.2.2 uint32_t wds_DataULongTlv::ulData

8.783 wds_DHCPLeaseOptTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t numOpt
- wds_DHCPOpt optList [30]
- uint8_t optListData [2048]

8.783.1 Detailed Description

DHCP lease option information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
numOpt	number of sets of wds_DHCPOpt
optList	option list
optListData	option list data

8.783.2 Field Documentation

8.783.2.1 uint8_t wds_DHCPLeaseOptTlv::numOpt

8.783.2.2 wds_DHCPOpt wds_DHCPLeaseOptTlv::optList[30]

8.783.2.3 uint8_t wds_DHCPLeaseOptTlv::optListData[2048]

8.783.2.4 uint8_t wds_DHCPLeaseOptTlv::TlvPresent

8.784 wds_DHCPLeaseStateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t leaseState

8.784.1 Detailed Description

DHCP lease state information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
leaseState	lease state • 0 - active, newly acquired • 1 - active, renewed • 2 - active, renewing • 3 - active, rebinding • 4 - inactive, expired • 5 - inactive, renew refused • 6 - inactive, rebind refused • 7 - inactive, other

8.784.2 Field Documentation

8.784.2.1 uint8_t wds_DHCPLeaseStateTlv::leaseState

8.784.2.2 uint8_t wds_DHCPLeaseStateTlv::TlvPresent

8.785 wds_DHCPOpt Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t * pOptVal

8.785.1 Detailed Description

DHCP option code information

Parameters

optCode	option code • values: – 0 - 255
optValLen	length of option code • values: – 0 - 255
pOptVal	option value

8.785.2 Field Documentation

8.785.2.1 uint8_t wds_DHCPOpt::optCode

8.785.2.2 uint8_t wds_DHCPOpt::optValLen

8.785.2.3 uint8_t* wds_DHCPOpt::pOptVal

8.786 wds_DHCPProfileIdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t profileType
- uint8_t profileId

8.786.1 Detailed Description

DHCP profile id TLV information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
profileType	identifying the type of the profile • 0 - 3GPP
profileId	index identifying the profile • 1-24 valid for 3GPP profile type (9x30 and onwards)

8.786.2 Field Documentation

8.786.2.1 uint8_t wds_DHCPProfileIdTlv::profileId

8.786.2.2 uint8_t wds_DHCPProfileIdTlv::profileType

8.786.2.3 uint8_t wds_DHCPProfileIdTlv::TlvPresent

8.787 wds_DHCPv4HWConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.787.1 Detailed Description

WDS SWI DHCPv4 HW Config Structure.

Parameters

hwType	• HW Type 1 - Ethernet 20 - Serial
chaddrlen	• chaddrlen
chaddr	• chaddr. Max size 16 bytes

8.787.2 Field Documentation

8.787.2.1 uint8_t wds_DHCPv4HWConfig::chaddr[16]

8.787.2.2 uint8_t wds_DHCPv4HWConfig::chaddrLen

8.787.2.3 uint8_t wds_DHCPv4HWConfig::hwType

8.788 wds_DHCPv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.788.1 Detailed Description

WDS SWI DHCPv4 Option Structure

Parameters

optCode	• Option code – 0 - 255
optValLen	• Option value length – 0 - 255
optVal	• Option value

8.788.2 Field Documentation

8.788.2.1 uint8_t wds_DHCPv4Option::optCode

8.788.2.2 uint8_t wds_DHCPv4Option::optVal[255]

8.788.2.3 uint8_t wds_DHCPv4Option::optValLen

8.789 wds_DHCPv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wds_DHCPv4Option * pOptList

8.789.1 Detailed Description

WDS SWI DHCPv4 Option List Structure

Parameters

numOpt	• number of options – 0 - 255
pOptList	• pointer to list of DHCP Options

8.789.2 Field Documentation

8.789.2.1 uint8_t wds_DHCPv4OptionList::numOpt

8.789.2.2 wds_DHCPv4Option* wds_DHCPv4OptionList::pOptList

8.790 wds_DHCPv4ProfileId Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileId

8.790.1 Detailed Description

WDS SWI DHCPv4 Profile Identifier Structure

Parameters

profileType	• 0 for 3GPP
profileId	• 1 to 24 for 3GPP profile

8.790.2 Field Documentation

8.790.2.1 uint8_t wds_DHCPv4ProfileId::profileId

8.790.2.2 uint8_t wds_DHCPv4ProfileId::profileType

8.791 wds_Domain Struct Reference

Data Fields

- uint16_t domainLen
- uint8_t domainName [256]

8.791.1 Detailed Description

This structure contains the DomainName Information

Parameters

domainLen	• length of the recieved Domain name
domainName	• Domain name(Max 256 characters)

8.791.2 Field Documentation

8.791.2.1 uint16_t wds_Domain::domainLen

8.791.2.2 uint8_t wds_Domain::domainName[256]

8.792 wds_DomainNameList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_Domain domain [10]

8.792.1 Detailed Description

This structure contains the DomainNameList Information

Parameters

numInstances	• Number of Domain name receieved
domain	• Domain name information(Max 10 Domain names)

8.792.2 Field Documentation

8.792.2.1 struct wds_Domain wds_DomainNameList::domain[10]

8.792.2.2 uint8_t wds_DomainNameList::numInstances

8.793 wds_GPRSQoS Struct Reference

Data Fields

- uint32_t precedenceClass
- uint32_t delayClass
- uint32_t reliabilityClass
- uint32_t peakThroughputClass
- uint32_t meanThroughputClass

8.793.1 Detailed Description

This structure contains the GPRS Quality Of Service Information

Parameters

precedence-Class	• Precedence class
delayClass	• Delay class
reliabilityClass	• Reliability class
peak-Throughput-Class	• Peak throughput class
mean-Throughput-Class	• Mean throughput class

8.793.2 Field Documentation

8.793.2.1 uint32_t wds_GPRSQoS::delayClass

8.793.2.2 uint32_t wds_GPRSQoS::meanThroughputClass

8.793.2.3 uint32_t wds_GPRSQoS::peakThroughputClass

8.793.2.4 uint32_t wds_GPRSQoS::precedenceClass

8.793.2.5 uint32_t wds_GPRSQoS::reliabilityClass

8.794 wds_IPv4AdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t IPv4Addr

8.794.1 Detailed Description

IPv4 address TLV information

Parameters

TlvPresent	a flag indicates whether TLV presented or not
IPv4Addr	IPv4 address

8.794.2 Field Documentation

8.794.2.1 uint32_t wds_IPv4AdTlv::IPv4Addr

8.794.2.2 uint8_t wds_IPv4AdTlv::TlvPresent

8.795 wds_IPV6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.795.1 Detailed Description

This structure contains the IPV6 Address Information

Parameters

IPV6PrefixLen	• Length of the received IPv6 address in no. of bits; can take value between 0 and 128 – 0xFF - Not Available
IPAddressV6	• IPv6 address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

8.795.2 Field Documentation

8.795.2.1 uint16_t wds_IPV6AddressInfo::IPAddressV6[8]

8.795.2.2 uint8_t wds_IPV6AddressInfo::IPV6PrefixLen

8.796 wds_IPV6GWAddressInfo Struct Reference

Data Fields

- uint8_t gwV6PrefixLen
- uint16_t gwAddressV6 [8]

8.796.1 Detailed Description

This structure contains the IPV6 Gateway Address Information

Parameters

gwV6PrefixLen	• Length of the received IPv6 Gateway address in no. of bits; can take value between 0 and 128
IPAddressV6	• IPv6 Gateway address(in network byte order); This is an 8-element array of 16 bit numbers, each of which is in big endian format.

8.796.2 Field Documentation

8.796.2.1 uint16_t wds_IPV6GWAddressInfo::gwAddressV6[8]

8.796.2.2 uint8_t wds_IPV6GWAddressInfo::gwV6PrefixLen

8.797 wds_PCSCFFQDNAddress Struct Reference

Data Fields

- uint16_t fqdnLen
- uint8_t fqdnAddr [256]

8.797.1 Detailed Description

This structure contains the PCSCFFQDNAddress Information

Parameters

fqdnLen	• length of the received FQDN address
fqdnAddr	• FQDN address(Max 256 characters)

8.797.2 Field Documentation

8.797.2.1 uint8_t wds_PCSCFFQDNAddress::fqdnAddr[256]

8.797.2.2 uint16_t wds_PCSCFFQDNAddress::fqdnLen

8.798 wds_PCSCFFQDNAddressList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_PCSCFFQDNAddress pcsfFQDNAddress [10]

8.798.1 Detailed Description

This structure contains the PCSCFFQDNAddressList Information

Parameters

numInstances	• Number of FQDN addresses received
pcsfFQDN-Address	• FQDN address information(Max 10 addresses)

8.798.2 Field Documentation

8.798.2.1 uint8_t wds_PCSCFFQDNAddressList::numInstances

8.798.2.2 struct wds_PCSCFFQDNAddress wds_PCSCFFQDNAddressList::pcsfFQDNAddress[10]

8.799 wds_PCSCFIPv4ServerAddressList Struct Reference

Data Fields

- uint8_t numInstances
- uint32_t pscsfIPv4Addr [64]

8.799.1 Detailed Description

This structure contains the PCSCFIPv4ServerAddressList Information

Parameters

numInstances	• number of address following
pscsfIPv4Addr	• P-CSCF IPv4 server addresses(Max 16 address, 4 bytes each)

8.799.2 Field Documentation

8.799.2.1 uint8_t wds_PCSCFIPv4ServerAddressList::numInstances

8.799.2.2 uint32_t wds_PCSCFIPv4ServerAddressList::pscsfIPv4Addr[64]

8.800 wds_ProfileIdentifier Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.800.1 Detailed Description

This structure contains the Profile Identifier Information

Parameters

profileType	• Identifies the type of profile 0x00 = 3GPP
profileIndex	• Index of profile whose settings were loaded prior to session parameter negotiation for the current call. If this TLV is not present, data call parameters are based on device default settings for each parameter

8.800.2 Field Documentation

8.800.2.1 uint8_t wds_ProfileIdentifier::profileIndex

8.800.2.2 uint8_t wds_ProfileIdentifier::profileType

8.801 wds_profileInfo Union Reference

Data Fields

- LibPackprofile_3GPP SlqsProfile3GPP
- LibPackprofile_3GPP2 SlqsProfile3GPP2

8.801.1 Detailed Description

This union consist of profile_3GPP and profile_3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile
SlqsProfile3GP-P2	3GPP2 profile

8.801.2 Field Documentation

8.801.2.1 LibPackprofile_3GPP wds_profileInfo::SlqsProfile3GPP

8.801.2.2 LibPackprofile_3GPP2 wds_profileInfo::SlqsProfile3GPP2

8.802 wds_TrStatInd Struct Reference

Data Fields

- uint8_t statsPeriod
- uint32_t statsMask

8.802.1 Detailed Description

This structure contains the information about the Transfer Statistics Indicator parameters.

Parameters

statsPeriod	• Period between transfer statistics reports. – 0 - Do not report – Other - Period between reports (seconds)
statsMask	• Requested statistic bit mask. – 0x00000001 - Tx packets OK – 0x00000002 - Rx packets OK – 0x00000004 - Tx packet errors – 0x00000008 - Rx packet errors – 0x00000010 - Tx overflows – 0x00000020 - Rx overflows – 0x00000040 - Tx bytes OK – 0x00000080 - Rx bytes OK • Each bit set causes the corresponding optional information to be sent in SLQSWdsEventReport-CallBack. • All unlisted bits are reserved for future use and must be set to zero.

8.802.2 Field Documentation

8.802.2.1 uint32_t wds_TrStatInd::statsMask

8.802.2.2 uint8_t wds_TrStatInd::statsPeriod

8.803 wds_UMTSMinQoS Struct Reference

Data Fields

- uint8_t trafficClass
- uint32_t maxUplinkBitrate
- uint32_t maxDownlinkBitrate
- uint32_t grntUplinkBitrate
- uint32_t grntDownlinkBitrate
- uint8_t qosDeliveryOrder
- uint32_t maxSDUSize
- uint8_t sduErrorRatio
- uint8_t resBerRatio
- uint8_t deliveryErrSDU
- uint32_t transferDelay
- uint32_t trafficPriority

8.803.1 Detailed Description

This structure contains the UMTS Quality Of Service Information

Parameters

trafficClass	• 0x00 - Subscribed • 0x01 - Conversational • 0x02 - Streaming • 0x03 - Interactive • 0x04 - Background
maxUplinkBitrate	• Maximum uplink bit rate in bits/sec
maxDownlink-Bitrate	• Maximum downlink bit rate in bits/sec
grntUplinkBitrate	• Guaranteed uplink bit rate in bits/sec
grntDownlink-Bitrate	• Guaranteed downlink bit rate in bits/sec
qosDelivery-Order	- Qos delivery order • 0x00 - Subscribe • 0x01 - Delivery order on • 0x02 - Delivery order off
maxSDUSize	• Maximum SDU size
sduErrorRatio	- SDU error ratio • Target value for fraction of SDUs lost or detected as erroneous. • 0x00 - Subscribe • 0x01 - 1∗10^(-2) • 0x02 - 7∗10^(-3) • 0x03 - 1∗10^(-3) • 0x04 - 1∗10^(-4) • 0x05 - 1∗10^(-5) • 0x06 - 1∗10^(-6) • 0x07 - 1∗10^(-1)
resBerRatio	- Residual bit error ratio • Target value for undetected bit error ratio in in the delivered SDUs. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)

deliveryErrSDU	- delivery of erroneous SDUs • Indicates whether SDUs detected as erroneous shall be delivered or not. • 0x00 - Subscribe • 0x01 - 5∗10^(-2) • 0x02 - 1∗10^(-2) • 0x03 - 5∗10^(-3) • 0x04 - 4∗10^(-3) • 0x05 - 1∗10^(-3) • 0x06 - 1∗10^(-4) • 0x07 - 1∗10^(-5) • 0x08 - 1∗10^(-6) • 0x09 - 1∗10^(-8)
transferDelay	- Transfer delay (ms) • Indicates the targeted time between a request to transfer an SDU at one SAP to its delivery at the other SAP in milliseconds.
trafficPriority	- Transfer handling priority • Specifies the relative importance for handling of SDUs that belong to the UMTS bearer, compared to the SDUs of other bearers.

8.803.2 Field Documentation

8.803.2.1 uint8_t wds_UMTSMinQoS::deliveryErrSDU

8.803.2.2 uint32_t wds_UMTSMinQoS::grntDownlinkBitrate

8.803.2.3 uint32_t wds_UMTSMinQoS::grntUplinkBitrate

8.803.2.4 uint32_t wds_UMTSMinQoS::maxDownlinkBitrate

8.803.2.5 uint32_t wds_UMTSMinQoS::maxSDUSize

8.803.2.6 uint32_t wds_UMTSMinQoS::maxUplinkBitrate

8.803.2.7 uint8_t wds_UMTSMinQoS::qosDeliveryOrder

8.803.2.8 uint8_t wds_UMTSMinQoS::resBerRatio

8.803.2.9 uint8_t wds_UMTSMinQoS::sduErrorRatio

8.803.2.10 uint8_t wds_UMTSMinQoS::trafficClass

8.803.2.11 uint32_t wds_UMTSMinQoS::trafficPriority

8.803.2.12 uint32_t wds_UMTSMinQoS::transferDelay

8.804 wdsDhcpv4HwConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.804.1 Detailed Description

Parameters

hwType	DHCP HW Type, examples: • 0 - Ethernet • 20 - Serial
chaddrLen	Length of chaddr field, examples: • 6 for Ethernet MAC address
chaddr	Client hardware address

8.804.2 Field Documentation

8.804.2.1 uint8_t wdsDhcpv4HwConfig::chaddr[16]

8.804.2.2 uint8_t wdsDhcpv4HwConfig::chaddrLen

8.804.2.3 uint8_t wdsDhcpv4HwConfig::hwType

8.805 wdsDhcpv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.805.1 Detailed Description

Parameters

optCode	Option code • 0 - 255
optValLen	Option value length • 0 - 255
optVal	Option Value

8.805.2 Field Documentation

8.805.2.1 uint8_t wdsDhcpv4Option::optCode

8.805.2.2 uint8_t wdsDhcpv4Option::optVal[255]

8.805.2.3 uint8_t wdsDhcpv4Option::optValLen

8.806 wdsDhcpv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wdsDhcpv4Option * pOptList

8.806.1 Detailed Description

Parameters

numOpt	number of options • 0 - 255
pOptList	pointer to list of DHCP Options

8.806.2 Field Documentation

8.806.2.1 uint8_t wdsDhcpv4OptionList::numOpt

8.806.2.2 wdsDhcpv4Option* wdsDhcpv4OptionList::pOptList

8.807 wdsDhcpv4ProfileId Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileId

8.807.1 Detailed Description

Parameters

profileType	profile type • 0 - 3GPP
profileId	profile index • index identifiying the profile 1-24 valid for 3GPP profile type (EM74xx and onwards)

8.807.2 Field Documentation

8.807.2.1 uint8_t wdsDhcpv4ProfileId::profileId

8.807.2.2 uint8_t wdsDhcpv4ProfileId::profileType

Chapter 9

File Documentation

9.1 apdoxypages.c File Reference

Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages.

Namespaces

- Tables

9.1.1 Detailed Description

Contains the module declaration for the Doxygen output. Also contains the content of the main page and related pages.

9.2 cat.h File Reference

Data Structures

- struct pack_cat_SetCATEventCallback_t
- struct unpack_cat_SetCATEventCallback_t
- struct cat_EventIDDataTlv
- struct cat_AlPhaIdentifierTlv
- struct cat_EventListTlv
- struct cat_RefreshTlv
- struct cat_EndProactiveSessionTlv
- union cat_currentCatEvent
- struct cat_commonEventTlv
- struct unpack_cat_SetCatEventCallback_ind_t
- struct pack_cat_CATSendEnvelopeCommand_t
- struct pack_cat_CATSendTerminalResponse_t

Macros

- #define CAT_EVENT_DATA_MAX_LENGTH 255
- #define CAN_COMMON_EVENT_TLV_NUMBER 11

Functions

- int pack_cat_SetCATEventCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_cat_SetCATEventCallback_t *reqArg)
- int unpack_cat_SetCATEventCallback (uint8_t *pResp, uint16_t respLen, unpack_cat_SetCATEventCallback_t *pOutput)
- int unpack_cat_SetCatEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_cat_SetCatEventCallback_ind_t *pOutput)
- int pack_cat_CATSendEnvelopeCommand (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_cat_CATSendEnvelopeCommand_t *reqArg)
- int unpack_cat_CATSendEnvelopeCommand (uint8_t *pResp, uint16_t respLen)
- int pack_cat_CATSendTerminalResponse (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_cat_CATSendTerminalResponse_t *reqArg)
- int unpack_cat_CATSendTerminalResponse (uint8_t *pResp, uint16_t respLen)

9.2.1 Macro Definition Documentation

9.2.1.1 #define CAN_COMMON_EVENT_TLV_NUMBER 11

9.2.1.2 #define CAT_EVENT_DATA_MAX_LENGTH 255

9.2.2 Function Documentation

9.2.2.1 int pack_cat_CATSendEnvelopeCommand (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_cat_CATSendEnvelopeCommand_t * *reqArg*)

Sends the envelope command to the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.2.2 int pack_cat_CATSendTerminalResponse (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_cat_CATSendTerminalResponse_t * *reqArg*)

Sends the terminal response to the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter •

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.2.3 int pack_cat_SetCATEventCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_cat_SetCATEventCallback_t * *reqArg*)

Enables/disables the CAT event callback pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.2.4 int unpack_cat_CATSendEnvelopeCommand (uint8_t * *pResp*, uint16_t *respLen*)

Sends the envelope command to the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.2.5 int unpack_cat_CATSendTerminalResponse (uint8_t * *pResp*, uint16_t *respLen*)

Sends the terminal response to the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.2.6 int unpack_cat_SetCATEventCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_cat_SetCATEventCallback_t * *pOutput*)

Enables/disables the CAT event callback unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.2.2.7 int unpack_cat_SetCatEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_cat_SetCatEvent-Callback_ind_t * *pOutput*)

CAT event indication unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3 common.h File Reference

Data Structures

- struct pack_qmi_t
- struct unpack_qmi_t

Macros

- #define SDU_HDR_LEN (3)
- #define MINREQBKLEN (2048)
- #define MSGID_AND_LEN (4)
- #define MSGID_DONT_CARE (0xffff)
- #define UNUSEDPARAM(x) (void)x
- #define DEAULT_LOC_TIMEOUT_IN_SEC 2
- #define SDK_VALIDATE_INPUT_PACK_PARAM(pCtx, pBuf, pLen)
- #define SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID(pCtx, pBuf, pLen)

Typedefs

- typedef void(∗ logger)(uint8_t lvl, const char ∗buff)

Enumerations

- enum eLOG_LEVEL {
  eLOG_INFO,
  eLOG_DEBUG,
  eLOG_WARN,
  eLOG_FATAL }
- enum eTimeout {
  eTIMEOUT_2_S = 2000,
  eTIMEOUT_5_S = 5000,
  eTIMEOUT_8_S = 8000,
  eTIMEOUT_10_S = 10000,
  eTIMEOUT_20_S = 20000,
  eTIMEOUT_30_S = 30000,
  eTIMEOUT_60_S = 60000,
  eTIMEOUT_300_S = 300000,
  eTIMEOUT_DEFAULT = eTIMEOUT_8_S }
- enum eQMI_SVC {
  eCTL,
  eWDS,
  eDMS,
  eNAS =3,
  eQOS,
  eSMS =5,
  ePDS =0x06,
  eVOICE =0x09,
  eCAT =0x0A,
  eUIM =0x0B,
  eLOC =0x10,
  eSAR =0x11,
  eTMD =0x18,
  eIMSA =0x21,
  eRMS =225,
  eSWIOMA =240,
  eSWIM2MCMD =243,
  eSWIOMAEXT =244,
  eSWILOC =246 }
- enum msgtype {
  eREQ =0,
  eRSP =2,
  eIND =4 }

Functions

- uint16_t helper_get_xid (uint8_t *qmi_resp)
- const char * helper_get_resp_ctx (uint8_t svc, uint8_t *pbuf, uint32_t len, unpack_qmi_t *pCtx)
- unsigned unpack_result_code_only (uint8_t *pMdmResp)
- int helper_set_log_func (logger func)
- void liteqmi_log (uint8_t lvl, const char *fmt,...)
- int helper_set_log_lvl (uint8_t lvl)
- void fill_sdu_hdr (pack_qmi_t *pCtx, uint8_t *pReqBuf)
- void fill_pack_ctx (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t svc, int timeout)
- char * get_version ()
- char * liteqmi_GetVersion ()
- void liteqmi_helper_decode7bitAsciiEncString (uint8_t *encoded_str, uint8_t encoded_len, uint8_t *decoded_str)

Variables

- logger glog
- uint8_t gloglvl

9.3.1 Macro Definition Documentation

9.3.1.1 #define DEAULT_LOC_TIMEOUT_IN_SEC 2

9.3.1.2 #define MINREQBKLEN (2048)

9.3.1.3 #define MSGID_AND_LEN (4)

9.3.1.4 #define MSGID_DONT_CARE (0xffff)

9.3.1.5 #define SDK_VALIDATE_INPUT_PACK_PARAM(*pCtx, pBuf, pLen*)

Value:

```
if ((pCtx == NULL) || (pBuf == NULL) || (pLen == NULL) ) \
        {\
        liteqmi_log(eLOG_DEBUG,"[  pack] %s parameter NULL\n",__func__); \
        return eQCWWAN_ERR_INVALID_ARG; \
        }
```

9.3.1.6 #define SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID(*pCtx, pBuf, pLen*)

Value:

```
if ((pCtx == NULL) || (pBuf == NULL) || (pLen == NULL) ) \
        {\
        liteqmi_log(eLOG_DEBUG,"[  pack] %s parameter NULL\n",__func__); \
        return eQCWWAN_ERR_INVALID_ARG; \
        } \
    if (pCtx->xid == 0) \
       return eQCWWAN_ERR_INVALID_XID; \
    pBuf[0] = eREQ; \
    pBuf[1] = pCtx->xid & 0xff; \
    pBuf[2] = pCtx->xid >> 8;
```

9.3.1.7 #define SDU_HDR_LEN (3)

9.3.1.8 #define UNUSEDPARAM(*x*) (void)x

9.3.2 Typedef Documentation

9.3.2.1 typedef void(∗ logger)(uint8_t lvl, const char ∗buff)

9.3.3 Enumeration Type Documentation

9.3.3.1 enum eLOG_LEVEL

log levels

Enumerator

eLOG_INFO

eLOG_DEBUG

eLOG_WARN

eLOG_FATAL

9.3.3.2 enum eQMI_SVC

qmi service

Enumerator

eCTL

eWDS

eDMS

eNAS

eQOS

eSMS

ePDS

eVOICE

eCAT

eUIM

eLOC

eSAR

eTMD

eIMSA

eRMS

eSWIOMA

eSWIM2MCMD

eSWIOMAEXT

eSWILOC

9.3.3.3 enum eTimeout

eTimeout

Enumerator

eTIMEOUT_2_S

eTIMEOUT_5_S

eTIMEOUT_8_S

eTIMEOUT_10_S

eTIMEOUT_20_S

eTIMEOUT_30_S

eTIMEOUT_60_S

eTIMEOUT_300_S

eTIMEOUT_DEFAULT

9.3.3.4 enum msgtype

qmi message type

Enumerator

eREQ

eRSP

eIND

9.3.4 Function Documentation

9.3.4.1 void fill_pack_ctx (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* uint8_t *svc,* int *timeout*)

9.3.4.2 void fill_sdu_hdr (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf*)

9.3.4.3 char* get_version ( )

Returns

version string

9.3.4.4 const char* helper_get_resp_ctx (uint8_t *svc,* uint8_t * *pbuf,* uint32_t *len,* unpack_qmi_t * *pCtx*)

extract msgid/xid/type from modem reply

Parameters

in	*svc*	qmi service
in	*pbuf*	qmi response/indication
in	*len*	response/indication length
out	*pCtx*	unpacked context

Returns

qmi message string

9.3.4.5 uint16_t helper_get_xid (uint8_t * *qmi_resp*)

9.3.4.6 int helper_set_log_func (logger *func*)

set log function

9.3.4.7 int helper_set_log_lvl (uint8_t *lvl*)

set log level

9.3.4.8 char* liteqmi_GetVersion ( )

Returns

version string

9.3.4.9 void liteqmi_helper_decode7bitAsciiEncString (uint8_t * *encoded_str,* uint8_t *encoded_len,* uint8_t * *decoded_str*)

decode 7bit encoded string

Parameters

in	*encoded_str*	7 bit encoded ASCII
in	*encoded_len*	encoded length
out	*decoded_str*	decoded ASCII string

9.3.4.10 void liteqmi_log (uint8_t *lvl,* const char * *fmt,* ...)

9.3.4.11 unsigned unpack_result_code_only (uint8_t * *pMdmResp*)

common handler for unpacking response with TLV type 0x02 only

9.3.5 Variable Documentation

9.3.5.1 logger glog

9.3.5.2 uint8_t gloglvl

9.4 dms.h File Reference

Data Structures

- struct unpack_dms_GetModelID_t
- struct unpack_dms_GetIMSI_t
- struct unpack_dms_GetFirmwareInfo_t
- struct unpack_dms_GetPower_t
- struct unpack_dms_GetSerialNumbers_t
- struct unpack_dms_GetHardwareRevision_t
- struct unpack_dms_SLQSGetBandCapability_t
- struct unpack_dms_GetDeviceCapabilities_t
- struct unpack_dms_GetFirmwareRevisions_t
- struct unpack_dms_GetFirmwareRevision_t

- struct unpack_dms_GetDeviceSerialNumbers_t
- struct unpack_dms_GetPRLVersion_t
- struct unpack_dms_GetNetworkTime_t
- struct unpack_dms_GetVoiceNumber_t
- struct unpack_dms_GetDeviceHardwareRev_t
- struct unpack_dms_GetFSN_t
- struct unpack_dms_GetDeviceCap_t
- struct pack_dms_SetPower_t
- struct unpack_dms_SetPower_t
- struct unpack_dms_GetBandCapability_t
- struct unpack_dms_GetUSBComp_t
- struct pack_dms_SetUSBComp_t
- struct unpack_dms_SetUSBComp_t
- struct pack_dms_SetCustFeature_t
- struct unpack_dms_SetCustFeature_t
- struct unpack_dms_GetCustFeature_t
- struct unpack_dms_SetFirmwarePreference_t
- struct unpack_dms_GetCrashAction_t
- struct pack_dms_SetCrashAction_t
- struct unpack_dms_SetCrashAction_t
- struct unpack_dms_GetDeviceMfr_t
- struct pack_dms_SetEventReport_t
- struct unpack_dms_SetEventReport_t
- struct dms_OperatingModeTlv
- struct dms_ActivationStatusTlv
- struct unpack_dms_SetEventReport_ind_t
- struct pack_dms_UIMGetICCID_t
- struct unpack_dms_UIMGetICCID_t
- struct pack_dms_SetCustFeaturesV2_t
- struct unpack_dms_SetCustFeaturesV2_t
- struct pack_dms_GetCustFeaturesV2_t
- struct DMSgetCustomInput
- struct DMScustSettingInfo
- struct DMScustSettingList
- struct DMSgetCustomFeatureV2
- struct unpack_dms_GetCustFeaturesV2_t
- struct unpack_dms_GetActivationState_t
- struct image_info_t
- struct unpack_dms_SLQSSwiGetFirmwareCurr_t
- struct pack_dms_SLQSSwiSetDyingGaspCfg_t
- struct unpack_dms_SLQSSwiSetDyingGaspCfg_t
- struct unpack_dms_SLQSSwiClearDyingGaspStatistics_t
- struct packgetDyingGaspStatistics
- struct unpack_dms_SLQSSwiGetDyingGaspStatistics_t
- struct packgetDyingGaspCfg
- struct unpack_dms_SLQSSwiGetDyingGaspCfg_t
- struct unpack_dms_SLQSDmsSwiGetResetInfo_t
- struct unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t
- struct pack_dms_SLQSDmsSwiIndicationRegister_t
- struct unpack_dms_SLQSDmsSwiIndicationRegister_t
- struct unpack_dms_SLQSSwiGetFwUpdateStatus_t
- struct unpack_dms_GetManufacturer_t
- struct unpack_dms_GetOfflineReason_t
- struct pack_dms_SetActivationStatusCallback_t
- struct unpack_dms_SetActivationStatusCallback_t

- struct pack_dms_UIMSetPINProtection_t
- struct unpack_dms_UIMSetPINProtection_t
- struct pack_dms_UIMUnblockPIN_t
- struct pack_dms_UIMVerifyPIN_t
- struct pack_dms_UIMChangePIN_t
- struct pack_dms_UIMGetControlKeyStatus_t
- struct unpack_dms_UIMGetControlKeyStatus_t
- struct unpack_dms_UIMGetPINStatus_t
- struct pack_dms_UIMSetControlKeyProtection_t
- struct unpack_dms_UIMSetControlKeyProtection_t
- struct pack_dms_UIMUnblockControlKey_t
- struct unpack_dms_UIMUnblockControlKey_t
- struct pack_dms_ResetToFactoryDefaults_t
- struct unpack_dms_ResetToFactoryDefaults_t
- struct pack_dms_ActivateAutomatic_t
- struct eriDataparams
- struct unpack_dms_SLQSGetERIFile_t
- struct unpack_dms_SLQSUIMGetState_t
- struct pack_dms_SLQSSwiGetCrashInfo_t
- struct crashInformation
- struct crashInfoParams
- struct unpack_dms_SLQSSwiGetCrashInfo_t
- struct unpack_dms_SLQSSwiGetHostDevInfo_t
- struct pack_dms_SLQSSwiSetHostDevInfo_t
- struct unpack_dms_SLQSSwiSetHostDevInfo_t
- struct unpack_dms_SLQSSwiGetOSInfo_t
- struct pack_dms_SLQSSwiSetOSInfo_t
- struct unpack_dms_SLQSSwiSetOSInfo_t
- struct unpack_dms_SLQSSwiGetSerialNoExt_t

Macros

- #define DMS_UINT8_MAX_STRING_SZ 255
- #define DMS_MAX_CUST_ID_LEN 64
- #define DMS_MAX_CUST_VALUE_LEN 8
- #define DMS_IMGDETAILS_LEN 16
- #define DMS_MAX_FWUPDATE_LOG_STR_SZ 255
- #define DMS_MAX_FWUPDATE_REF_STR_SZ 15
- #define DMS_SLQSFWINFO_MODELID_SZ 20
- #define DMS_SLQSFWINFO_BOOTVERSION_SZ 85
- #define DMS_SLQSFWINFO_APPVERSION_SZ 85
- #define DMS_SLQSFWINFO_SKU_SZ 15
- #define DMS_SLQSFWINFO_PACKAGEID_SZ 85
- #define DMS_SLQSFWINFO_CARRIER_SZ 20
- #define DMS_SLQSFWINFO_PRIVERSION_SZ 16
- #define DMS_SLQSFWINFO_CUR_CARR_NAME 17
- #define DMS_SLQSFWINFO_CUR_CARR_REV 13
- #define MAX_BUILD_ID_LEN 255
- #define UNIQUE_ID_LEN 16
- #define SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160
- #define SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20
- #define SPC_SIZE 6
- #define CK_MAX_SIZE 8
- #define ACT_CODE_MAX_SIZE 81

- #define ERI_DATA_MAX_SIZE 1024
- #define MEID_MAX_SIZE 8
- #define DMS_PM_ONLINE 0x00 /∗ Online ∗/
- #define DMS_PM_LOW 0x01 /∗ Low Power ∗/
- #define DMS_PM_FACTORY 0x02 /∗ Factory Test Mode ∗/
- #define DMS_PM_OFFLINE 0x03 /∗ Offline ∗/
- #define DMS_PM_RESET 0x04 /∗ Reset ∗/
- #define DMS_PM_SHUT_DOWN 0x05 /∗ Shut Down ∗/
- #define DMS_PM_PERSISTENT_LOW 0x06 /∗ Persistent Low Power ∗/
- #define DMS_SET_REPORT_ENABLE 1
- #define DMS_SET_REPORT_DISABLE 0
- #define DMS_SWI_SET_IND_ENABLE 1
- #define DMS_SWI_SET_IND_DISABLE 0

Functions

- int pack_dms_GetIMSI (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetIMSI (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetIMSI_t ∗pOutput)
- int pack_dms_GetModelID (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetModelID (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetModelID_t ∗pOutput)
- int pack_dms_GetFirmwareInfo (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetFirmwareInfo (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetFirmwareInfo_t ∗pOutput)
- int pack_dms_GetPower (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetPower (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetPower_t ∗pOutput)
- int pack_dms_GetSerialNumbers (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetSerialNumbers (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetSerialNumbers_t ∗pOutput)
- int pack_dms_GetHardwareRevision (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetHardwareRevision (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetHardwareRevision_t ∗pOutput)
- int pack_dms_SLQSGetBandCapability (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_SLQSGetBandCapability (uint8_t ∗pResp, uint16_t respLen, unpack_dms_SLQSGetBandCapability_t ∗pOutput)
- int pack_dms_GetDeviceCapabilities (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetDeviceCapabilities (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetDeviceCapabilities_t ∗pOutput)
- int pack_dms_GetFirmwareRevisions (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetFirmwareRevisions (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetFirmwareRevisions_t ∗pOutput)
- int pack_dms_GetFirmwareRevision (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetFirmwareRevision (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetFirmwareRevision_t ∗pOutput)
- int pack_dms_GetDeviceSerialNumbers (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetDeviceSerialNumbers (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetDeviceSerialNumbers_t ∗pOutput)
- int pack_dms_GetPRLVersion (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetPRLVersion (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetPRLVersion_t ∗pOutput)
- int pack_dms_GetNetworkTime (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetNetworkTime (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetNetworkTime_t ∗pOutput)
- int pack_dms_GetVoiceNumber (pack_qmi_t ∗pCtx, uint8_t ∗pReqBuf, uint16_t ∗pLen, void ∗reqArg)
- int unpack_dms_GetVoiceNumber (uint8_t ∗pResp, uint16_t respLen, unpack_dms_GetVoiceNumber_t ∗pOutput)

- int pack_dms_GetDeviceHardwareRev (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetDeviceHardwareRev (uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceHardwareRev_t *pOutput)
- int pack_dms_GetFSN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetFSN (uint8_t *pResp, uint16_t respLen, unpack_dms_GetFSN_t *pOutput)
- int pack_dms_GetDeviceCap (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetDeviceCap (uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceCap_t *pOutput)
- int pack_dms_SetPower (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetPower_t *reqArg)
- int unpack_dms_SetPower (uint8_t *pResp, uint16_t respLen, unpack_dms_SetPower_t *pOutput)
- int pack_dms_GetBandCapability (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetBandCapability (uint8_t *pResp, uint16_t respLen, unpack_dms_GetBandCapability_t *pOutput)
- int pack_dms_GetUSBComp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetUSBComp (uint8_t *pResp, uint16_t respLen, unpack_dms_GetUSBComp_t *pOutput)
- int pack_dms_SetUSBComp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetUSBComp_t *reqArg)
- int unpack_dms_SetUSBComp (uint8_t *pResp, uint16_t respLen, unpack_dms_SetUSBComp_t *pOutput)
- int pack_dms_SetCustFeature (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCustFeature_t *reqArg)
- int unpack_dms_SetCustFeature (uint8_t *pResp, uint16_t respLen, unpack_dms_SetCustFeature_t *pOutput)
- int pack_dms_GetCustFeature (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetCustFeature (uint8_t *pResp, uint16_t respLen, unpack_dms_GetCustFeature_t *pOutput)
- int pack_dms_SetFirmwarePreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_SetFirmwarePreference (uint8_t *pResp, uint16_t respLen, unpack_dms_SetFirmwarePreference_t *pOutput)
- int pack_dms_GetCrashAction (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetCrashAction (uint8_t *pResp, uint16_t respLen, unpack_dms_GetCrashAction_t *pOutput)
- int pack_dms_SetCrashAction (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCrashAction_t reqArg)
- int unpack_dms_SetCrashAction (uint8_t *pResp, uint16_t respLen, unpack_dms_SetCrashAction_t *pOutput)
- int pack_dms_GetDeviceMfr (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_dms_GetDeviceMfr (uint8_t *pResp, uint16_t respLen, unpack_dms_GetDeviceMfr_t *pOutput)
- int pack_dms_SetEventReport (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetEventReport_t *reqArg)
- int unpack_dms_SetEventReport (uint8_t *pResp, uint16_t respLen, unpack_dms_SetEventReport_t *pOutput)
- int unpack_dms_SetEventReport_ind (uint8_t *pResp, uint16_t respLen, unpack_dms_SetEventReport_ind_t *pOutput)
- int pack_dms_UIMGetICCID (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMGetICCID_t *reqArg)
- int unpack_dms_UIMGetICCID (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetICCID_t *pOutput)
- int pack_dms_SetCustFeaturesV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetCustFeaturesV2_t *reqArg)
- int unpack_dms_SetCustFeaturesV2 (uint8_t *pResp, uint16_t respLen, unpack_dms_SetCustFeaturesV2_t *pOutput)
- int pack_dms_GetCustFeaturesV2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_GetCustFeaturesV2_t *reqArg)
- int unpack_dms_GetCustFeaturesV2 (uint8_t *pResp, uint16_t respLen, unpack_dms_GetCustFeaturesV2_t *pOutput)
- int pack_dms_GetActivationState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)

- int unpack_dms_GetActivationState (uint8_t *pResp, uint16_t respLen, unpack_dms_GetActivationState_t *pOutput)
- int pack_dms_SLQSSwiGetFirmwareCurr (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetFirmwareCurr (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetFirmwareCurr_t *pOutput)
- int pack_dms_SLQSSwiSetDyingGaspCfg (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetDyingGaspCfg_t *reqArg)
- int unpack_dms_SLQSSwiSetDyingGaspCfg (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetDyingGaspCfg_t *pOutput)
- int pack_dms_SLQSSwiClearDyingGaspStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiClearDyingGaspStatistics (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiClearDyingGaspStatistics_t *pOutput)
- int pack_dms_SLQSSwiGetDyingGaspStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetDyingGaspStatistics (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspStatistics_t *pOutput)
- int pack_dms_SLQSSwiGetDyingGaspCfg (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetDyingGaspCfg (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetDyingGaspCfg_t *pOutput)
- int pack_dms_SLQSDmsSwiGetResetInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSDmsSwiGetResetInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_t *pOutput)
- int unpack_dms_SLQSDmsSwiGetResetInfo_Ind (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t *pOutput)
- int pack_dms_SLQSDmsSwiIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSDmsSwiIndicationRegister_t *reqArg)
- int unpack_dms_SLQSDmsSwiIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSDmsSwiIndicationRegister_t *pOutput)
- int pack_dms_SLQSSwiGetFwUpdateStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetFwUpdateStatus (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetFwUpdateStatus_t *pOutput)
- int pack_dms_GetManufacturer (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_GetManufacturer (uint8_t *pResp, uint16_t respLen, unpack_dms_GetManufacturer_t *pOutput)
- int pack_dms_GetOfflineReason (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_GetOfflineReason (uint8_t *pResp, uint16_t respLen, unpack_dms_GetOfflineReason_t *pOutput)
- int pack_dms_SetActivationStatusCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SetActivationStatusCallback_t *reqArg)
- int unpack_dms_SetActivationStatusCallback (uint8_t *pResp, uint16_t respLen, unpack_dms_SetActivationStatusCallback_t *pOutput)
- int pack_dms_UIMSetPINProtection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMSetPINProtection_t *pReq)
- int unpack_dms_UIMSetPINProtection (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)
- int pack_dms_UIMUnblockPIN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMUnblockPIN_t *pReq)
- int unpack_dms_UIMUnblockPIN (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)
- int pack_dms_UIMVerifyPIN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMVerifyPIN_t *pReq)
- int unpack_dms_UIMVerifyPIN (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)
- int pack_dms_UIMChangePIN (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMChangePIN_t *pReq)
- int unpack_dms_UIMChangePIN (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetPINProtection_t *pOutput)

- int pack_dms_UIMGetControlKeyStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMGetControlKeyStatus_t *pReq)
- int unpack_dms_UIMGetControlKeyStatus (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetControlKeyStatus_t *pOutput)
- int pack_dms_UIMGetPINStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_UIMGetPINStatus (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMGetPINStatus_t *pOutput)
- int pack_dms_UIMSetControlKeyProtection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMSetControlKeyProtection_t *pReq)
- int unpack_dms_UIMSetControlKeyProtection (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMSetControlKeyProtection_t *pOutput)
- int pack_dms_UIMUnblockControlKey (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_UIMUnblockControlKey_t *pReq)
- int unpack_dms_UIMUnblockControlKey (uint8_t *pResp, uint16_t respLen, unpack_dms_UIMUnblockControlKey_t *pOutput)
- int pack_dms_ResetToFactoryDefaults (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ResetToFactoryDefaults_t *pReq)
- int unpack_dms_ResetToFactoryDefaults (uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)
- int pack_dms_ValidateSPC (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ResetToFactoryDefaults_t *pReq)
- int unpack_dms_ValidateSPC (uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)
- int pack_dms_ActivateAutomatic (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_ActivateAutomatic_t *pReq)
- int unpack_dms_ActivateAutomatic (uint8_t *pResp, uint16_t respLen, unpack_dms_ResetToFactoryDefaults_t *pOutput)
- int pack_dms_SLQSGetERIFile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSGetERIFile (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSGetERIFile_t *pOutput)
- int pack_dms_SLQSUIMGetState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSUIMGetState (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSUIMGetState_t *pOutput)
- int pack_dms_SLQSSwiGetCrashInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiGetCrashInfo_t *pReq)
- int unpack_dms_SLQSSwiGetCrashInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetCrashInfo_t *pOutput)
- int pack_dms_SLQSSwiGetHostDevInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetHostDevInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetHostDevInfo_t *pOutput)
- int pack_dms_SLQSSwiSetHostDevInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetHostDevInfo_t *pReq)
- int unpack_dms_SLQSSwiSetHostDevInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetHostDevInfo_t *pOutput)
- int pack_dms_SLQSSwiGetOSInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetOSInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetOSInfo_t *pOutput)
- int pack_dms_SLQSSwiSetOSInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_dms_SLQSSwiSetOSInfo_t *pReq)
- int unpack_dms_SLQSSwiSetOSInfo (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiSetOSInfo_t *pOutput)
- int pack_dms_SLQSSwiGetSerialNoExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_dms_SLQSSwiGetSerialNoExt (uint8_t *pResp, uint16_t respLen, unpack_dms_SLQSSwiGetSerialNoExt_t *pOutput)

9.4.1 Macro Definition Documentation

9.4.1.1 #define ACT_CODE_MAX_SIZE 81

9.4.1.2 #define CK_MAX_SIZE 8

9.4.1.3 #define DMS_IMGDETAILS_LEN 16

9.4.1.4 #define DMS_MAX_CUST_ID_LEN 64

9.4.1.5 #define DMS_MAX_CUST_VALUE_LEN 8

9.4.1.6 #define DMS_MAX_FWUPDATE_LOG_STR_SZ 255

9.4.1.7 #define DMS_MAX_FWUPDATE_REF_STR_SZ 15

9.4.1.8 #define DMS_PM_FACTORY 0x02 /∗ Factory Test Mode ∗/

9.4.1.9 #define DMS_PM_LOW 0x01 /∗ Low Power ∗/

9.4.1.10 #define DMS_PM_OFFLINE 0x03 /∗ Offline ∗/

9.4.1.11 #define DMS_PM_ONLINE 0x00 /∗ Online ∗/

9.4.1.12 #define DMS_PM_PERSISTENT_LOW 0x06 /∗ Persistent Low Power ∗/

9.4.1.13 #define DMS_PM_RESET 0x04 /∗ Reset ∗/

9.4.1.14 #define DMS_PM_SHUT_DOWN 0x05 /∗ Shut Down ∗/

9.4.1.15 #define DMS_SET_REPORT_DISABLE 0

9.4.1.16 #define DMS_SET_REPORT_ENABLE 1

9.4.1.17 #define DMS_SLQSFWINFO_APPVERSION_SZ 85

9.4.1.18 #define DMS_SLQSFWINFO_BOOTVERSION_SZ 85

9.4.1.19 #define DMS_SLQSFWINFO_CARRIER_SZ 20

9.4.1.20 #define DMS_SLQSFWINFO_CUR_CARR_NAME 17

9.4.1.21 #define DMS_SLQSFWINFO_CUR_CARR_REV 13

9.4.1.22 #define DMS_SLQSFWINFO_MODELID_SZ 20

9.4.1.23 #define DMS_SLQSFWINFO_PACKAGEID_SZ 85

9.4.1.24 #define DMS_SLQSFWINFO_PRIVERSION_SZ 16

9.4.1.25 #define DMS_SLQSFWINFO_SKU_SZ 15

9.4.1.26 #define DMS_SWI_SET_IND_DISABLE 0

9.4.1.27 #define DMS_SWI_SET_IND_ENABLE 1

9.4.1.28 #define DMS_UINT8_MAX_STRING_SZ 255

9.4.1.29 #define ERI_DATA_MAX_SIZE 1024

9.4.1.30 #define MAX_BUILD_ID_LEN 255

9.4.1.31 #define MEID_MAX_SIZE 8

9.4.1.32 #define SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160

9.4.1.33 #define SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20

9.4.1.34 #define SPC_SIZE 6

9.4.1.35 #define UNIQUE_ID_LEN 16

9.4.2 Function Documentation

9.4.2.1 int pack_dms_ActivateAutomatic (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ActivateAutomatic_t * *pReq*)

Requests the device to perform automatic service activation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.2 int pack_dms_GetActivationState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Activation State pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.3 int pack_dms_GetBandCapability (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Band Capability pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.4 int pack_dms_GetCrashAction (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Crash Action pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.5 int pack_dms_GetCustFeature (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Custom Feature pack. This API is deprecated for EM74xx/MC74xx, please use pack_dms_GetCustFeatures-V2() instead for EM74xx/MC74xx.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.6 int pack_dms_GetCustFeaturesV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_GetCustFeaturesV2_t * *reqArg*)

Get Custom Feature pack This function queries the modem for a list of supported features. This function is for firmware version 2.0 and newer. Currently supported Customization features:

- GPIOSARENABLE
- GPSSEL
- IMSWITCHHIDE
- IPV6ENABLE
- WAKEHOSTEN

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.7 int pack_dms_GetDeviceCap (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Device Capabilities pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.8 int pack_dms_GetDeviceCapabilities (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get device capability pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.9 int pack_dms_GetDeviceHardwareRev (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Hardware Revision of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.10 int pack_dms_GetDeviceMfr (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Device Manufacture Name pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.11 int pack_dms_GetDeviceSerialNumbers (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)

Get Device Serial Number pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.12 int pack_dms_GetFirmwareInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)

Get firmware info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.13 int pack_dms_GetFirmwareRevision (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, void * reqArg)

Get Firmware Revision pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.14 int pack_dms_GetFirmwareRevisions (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Firmware Revisions pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.15 int pack_dms_GetFSN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get FSN pack. This API get the Factory Sequence Number of the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.16 int pack_dms_GetHardwareRevision (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get hardware revision of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.17 int pack_dms_GetIMSI (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get IMSI pack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_-xx_xx and all EM74xx firmware versions. Please use pack_uim_ReadTransparent()(EF ID: 3F00 7F20 6F07 for 2G card and 3F00 7FFF 6F07 for 3G card) instead for new firmware versions and new modules.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.18 int pack_dms_GetManufacturer (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get device manufacturer information pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.19 int pack_dms_GetModelID (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get model id pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.20 int pack_dms_GetNetworkTime (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Network Time pack Returns the current time of the device based on the value supported by the network.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.21 int pack_dms_GetOfflineReason (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get operating mode offline reason pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.22 int pack_dms_GetPower (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get operating mode of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.23 int pack_dms_GetPRLVersion (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get PRL Versions pack. Return version of the active Preferred Roaming List (PRL) in use by the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.24 int pack_dms_GetSerialNumbers (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get serial numbers pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.25 int pack_dms_GetUSBComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get USB Comp pack This API queries the modem's USB interface configuration and supported configuration parameters.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.26 int pack_dms_GetVoiceNumber (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Voice Number pack. Returns the voice number in use by the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.27 int pack_dms_ResetToFactoryDefaults (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ResetToFactoryDefaults_t * *pReq*)

Resets to default factory settings of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.28 int pack_dms_SetActivationStatusCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetActivationStatusCallback_t * *reqArg*)

Set activation status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.29 int pack_dms_SetCrashAction (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetCrashAction_t *reqArg*)

Set Crash Action pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.30 int pack_dms_SetCustFeature (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetCustFeature_t * *reqArg*)

Set Custom Feature pack. This API is deprecated for EM74xx/MC74xx, please use pack_dms_SetCustFeatures-V2() instead for EM74xx/MC74xx.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.31 int pack_dms_SetCustFeaturesV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetCustFeaturesV2_t * *reqArg*)

Set Cust Features pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.32 int pack_dms_SetEventReport (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetEventReport_t * *reqArg*)

Set Event Report pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.33 int pack_dms_SetFirmwarePreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Set Firmware Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.34 int pack_dms_SetPower (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetPower_t * *reqArg*)

Set Power pack Sets the operating mode of the device.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.35 int pack_dms_SetUSBComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetUSBComp_t * *reqArg*)

Set USB Comp pack This API is used to change the modem's USB interface configuration thus allowing a device to have multiple USB compositions. Devices will, by default, be configured to support a minimal set of interfaces to reduce end user modem installation time. Developers and some customers, however, require access to a custom set of interfaces. A reset is required for any change in the USB composition to take effect.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.36 int pack_dms_SLQSDmsSwiGetResetInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get reset info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.37 int pack_dms_SLQSDmsSwiIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSDmsSwiIndicationRegister_t * *reqArg*)

Set the registration state for different indication pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM/MC74xx onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.38 int pack_dms_SLQSGetBandCapability (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get band capability of the device pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter(NULL)

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.39 int pack_dms_SLQSGetERIFile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set UIM Get ERI file pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.40 int pack_dms_SLQSSwiClearDyingGaspStatistics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Clear Dying GASP Statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.41 int pack_dms_SLQSSwiGetCrashInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSSwiGetCrashInfo_t * *pReq*)

To Get the Crash Information from the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.42 int pack_dms_SLQSSwiGetDyingGaspCfg (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Dying GASP Config pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.43 int pack_dms_SLQSSwiGetDyingGaspStatistics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Dying GASP Statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.44 int pack_dms_SLQSSwiGetFirmwareCurr (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get currently active image pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.45 int pack_dms_SLQSSwiGetFwUpdateStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get Firmware Update status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.46 int pack_dms_SLQSSwiGetHostDevInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get host dev information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.47 int pack_dms_SLQSSwiGetOSInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To Get Host OS Info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.48 int pack_dms_SLQSSwiGetSerialNoExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To get serial number extension pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.49 int pack_dms_SLQSSwiSetDyingGaspCfg (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSSwiSetDyingGaspCfg_t * *reqArg*)

Set Dying GASP Config pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.50 int pack_dms_SLQSSwiSetHostDevInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSSwiSetHostDevInfo_t * *pReq*)

To set Host Dev Info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.51 int pack_dms_SLQSSwiSetOSInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSSwiSetOSInfo_t * *pReq*)

To set Host OS Info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.52 int pack_dms_SLQSUIMGetState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Returns the UIM state pack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API unpack_uim_GetCardStatus/ unpack_uim_GetCardStatus for new firmware versions and new modules

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.53 int pack_dms_UIMChangePIN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMChangePIN_t * *pReq*)

To set UIM change PIN pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.54 int pack_dms_UIMGetControlKeyStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMGetControlKeyStatus_t * *pReq*)

To set UIM Get Control Key Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.55 int pack_dms_UIMGetICCID (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMGetICCID_t * *reqArg*)

Packs the UIMGetICCID response message to a user-provided response structure. This API is deprecated on MC73xx/EM73xx modules. Since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use pack_uim_ReadTransparent()(EF ID: 3F00 2FE2) instead for new firmware versions and new modules.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.56 int pack_dms_UIMGetPINStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set UIM Get PIN Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.57 int pack_dms_UIMSetControlKeyProtection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMSetControlKeyProtection_t * *pReq*)

To set UIM Set Control Key Protection pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.58 int pack_dms_UIMSetPINProtection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMSetPINProtection_t * *pReq*)

To set UIM PIN protection pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.59 int pack_dms_UIMUnblockControlKey (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMUnblockControlKey_t * *pReq*)

To set UIM Unblock Control Key pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.60 int pack_dms_UIMUnblockPIN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMUnblockPIN_t * *pReq*)

To set UIM unblock PIN pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.61 int pack_dms_UIMVerifyPIN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_UIMVerifyPIN_t * *pReq*)

To set UIM verify PIN pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.62 int pack_dms_ValidateSPC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ResetToFactoryDefaults_t * *pReq*)

Requests the device to perform automatic service activation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
out	*pReq*	requeset parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.63 int unpack_dms_ActivateAutomatic (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactory-Defaults_t * *pOutput*)

Requests the device to perform automatic service activation unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.64 int unpack_dms_GetActivationState (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetActivationState_t * *pOutput*)

Get Activation State unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.65 int unpack_dms_GetBandCapability (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetBandCapability_t * *pOutput*)

Get Band Capabilities unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.66 int unpack_dms_GetCrashAction (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCrashAction_t * *pOutput*)

Get Crash Action unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.67 int unpack_dms_GetCustFeature (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCustFeature_t * *pOutput*)

Get Custom Feature unpack. This API is deprecated for EM74xx/MC74xx, please use unpack_dms_GetCustFeaturesV2() instead for EM74xx/MC74xx.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.68 int unpack_dms_GetCustFeaturesV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCustFeaturesV2_t * *pOutput*)

Get Custom Feature unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.69 int unpack_dms_GetDeviceCap (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceCap_t * *pOutput*)

Get Device Capabilities unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.70 int unpack_dms_GetDeviceCapabilities (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceCapabilities_t * *pOutput*)

Get device capability unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.71 int unpack_dms_GetDeviceHardwareRev (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceHardwareRev_t * *pOutput*)

Get Hardware Revision of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.72 int unpack_dms_GetDeviceMfr (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceMfr_t * *pOutput*)

Get Device Manufacture Name unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.73 int unpack_dms_GetDeviceSerialNumbers (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceSerialNumbers_t * *pOutput*)

Get Device Serial Number unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.74 int unpack_dms_GetFirmwareInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmwareInfo_t * *pOutput*)

Get firmware info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.75 int unpack_dms_GetFirmwareRevision (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmwareRevision_t * *pOutput*)

Get Firmware Revision unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.76 int unpack_dms_GetFirmwareRevisions (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmwareRevisions_t * *pOutput*)

Get Firmware Revisions unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.77 int unpack_dms_GetFSN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFSN_t * *pOutput*)

Get FSN unpack This API get the Factory Sequence Number of the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.78 int unpack_dms_GetHardwareRevision (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetHardwareRevision_t * *pOutput*)

Get hardware revision of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.79 int unpack_dms_GetIMSI (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetIMSI_t * *pOutput*)

Get IMSI unpack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx-_xx_xx and all EM74xx firmware versions. Please use unpack_uim_ReadTransparent()(EF ID: 3F00 7F20 6F07 for 2G card and 3F00 7FFF 6F07 for 3G card) instead for new firmware versions and new modules.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.80 int unpack_dms_GetManufacturer (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetManufacturer_t * *pOutput*)

To get device manufacturer information unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.81 int unpack_dms_GetModelID (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetModelID_t * *pOutput*)

Get model id unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.82 int unpack_dms_GetNetworkTime (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetNetworkTime_t * *pOutput*)

Get Network Time unpack. Returns the current time of the device based on the value supported by the network.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.83 int unpack_dms_GetOfflineReason (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetOfflineReason_t * *pOutput*)

To get operating mode offline reason unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.84 int unpack_dms_GetPower (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetPower_t * *pOutput*)

Get operating mode of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.85 int unpack_dms_GetPRLVersion (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetPRLVersion_t * *pOutput*)

Get PRL Versions unpack Return version of the active Preferred Roaming List (PRL) in use by the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.86 int unpack_dms_GetSerialNumbers (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetSerialNumbers_t * *pOutput*)

Get serial numbers unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.87 int unpack_dms_GetUSBComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetUSBComp_t * *pOutput*)

Get USB Comp unpack This API queries the modem's USB interface configuration and supported configuration parameters.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.88 int unpack_dms_GetVoiceNumber (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetVoiceNumber_t * *pOutput*)

Get Voice Number unpack Returns the voice number in use by the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.89 int unpack_dms_ResetToFactoryDefaults (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactory-Defaults_t * *pOutput*)

Resets to default factory settings of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.90 int unpack_dms_SetActivationStatusCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetActivationStatusCallback_t * *pOutput*)

Set Activation status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.91 int unpack_dms_SetCrashAction (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetCrashAction_t * *pOutput*)

Set Crash Action unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response. Not used

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.92 int unpack_dms_SetCustFeature (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetCustFeature_t * *pOutput*)

Set Custom Feature unpack. This API is deprecated for EM74xx/MC74xx, please use unpack_dms_SetCustFeaturesV2() instead for EM74xx/MC74xx.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.93 int unpack_dms_SetCustFeaturesV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetCustFeaturesV2_t * *pOutput*)

Set Cust features unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.94 int unpack_dms_SetEventReport (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetEventReport_t * *pOutput*)

Set Event Report unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.95 int unpack_dms_SetEventReport_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetEventReport_ind_t * *pOutput*)

Event Report Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.96 int unpack_dms_SetFirmwarePreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetFirmware-Preference_t * *pOutput*)

Set Firmware Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.97 int unpack_dms_SetPower (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetPower_t * *pOutput*)

Set Power unpack. Sets the operating mode of the device.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.98 int unpack_dms_SetUSBComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetUSBComp_t * *pOutput*)

Set USB Comp unpack This API is used to change the modem's USB interface configuration thus allowing a device to have multiple USB compositions. Devices will, by default, be configured to support a minimal set of interfaces to reduce end user modem installation time. Developers and some customers, however, require access to a custom set of interfaces. A reset is required for any change in the USB composition to take effect.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.99 int unpack_dms_SLQSDmsSwiGetResetInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiGetResetInfo_t * *pOutput*)

To get reset info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.100 int unpack_dms_SLQSDmsSwiGetResetInfo_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t * *pOutput*)

DMS reset info Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM/MC74xx onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.101 int unpack_dms_SLQSDmsSwiIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiIndicationRegister_t * *pOutput*)

Set the registration state for different indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

support EM/MC74xx onwards

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.102 int unpack_dms_SLQSGetBandCapability (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSGetBandCapability_t * *pOutput*)

Get band capability of the device unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.103 int unpack_dms_SLQSGetERIFile (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSGetERIFile_t * *pOutput*)

To get UIM Get ERI file unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.104 int unpack_dms_SLQSSwiClearDyingGaspStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiClearDyingGaspStatistics_t * *pOutput*)

Clear Dying GASP Statistics unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.105 int unpack_dms_SLQSSwiGetCrashInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetCrashInfo_t * *pOutput*)

To Get the Crash Information from the device unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.106 int unpack_dms_SLQSSwiGetDyingGaspCfg (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetDyingGaspCfg_t * *pOutput*)

Get Dying GASP Config unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.107 int unpack_dms_SLQSSwiGetDyingGaspStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetDyingGaspStatistics_t * *pOutput*)

Get Dying GASP Statistics unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.108 int unpack_dms_SLQSSwiGetFirmwareCurr (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetFirmwareCurr_t * *pOutput*)

Get currently active image unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.109 int unpack_dms_SLQSSwiGetFwUpdateStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetFwUpdateStatus_t * *pOutput*)

To get Firmware Update status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.110 int unpack_dms_SLQSSwiGetHostDevInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetHostDevInfo_t * *pOutput*)

To get host dev information unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.111 int unpack_dms_SLQSSwiGetOSInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetOSInfo_t * *pOutput*)

To get Host OS Info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.112 int unpack_dms_SLQSSwiGetSerialNoExt (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetSerialNoExt_t * *pOutput*)

To get serial number extension unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.113 int unpack_dms_SLQSSwiSetDyingGaspCfg (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetDyingGaspCfg_t * *pOutput*)

Set Dying GASP Config unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.114 int unpack_dms_SLQSSwiSetHostDevInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetHostDevInfo_t * *pOutput*)

To set Host Dev Info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.115 int unpack_dms_SLQSSwiSetOSInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetOSInfo_t * *pOutput*)

To set Host OS Info unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.116 int unpack_dms_SLQSUIMGetState (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSUIMGetState_t * *pOutput*)

Returns the UIM state unpack. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API unpack_uim_GetCardStatus / unpack_uim_GetCardStatus for new firmware versions and new modules

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.117 int unpack_dms_UIMChangePIN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM change PIN unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.118 int unpack_dms_UIMGetControlKeyStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMGetControlKeyStatus_t * *pOutput*)

To get UIM Get ControlKeyStatus unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.119 int unpack_dms_UIMGetICCID (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMGetICCID_t * *pOutput*)

Unpacks the UIMGetICCID response message to a user-provided response structure.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.120 int unpack_dms_UIMGetPINStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMGetPINStatus_t * *pOutput*)

To get UIM Get PIN Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.121 int unpack_dms_UIMSetControlKeyProtection (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetControlKeyProtection_t * *pOutput*)

To get UIM Set Control Key Protection unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.122 int unpack_dms_UIMSetPINProtection (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM Set PIN protection unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.123 int unpack_dms_UIMUnblockControlKey (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMUnblockControlKey_t * *pOutput*)

To get UIM Unblock Control Key unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.124 int unpack_dms_UIMUnblockPIN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM unblock PIN unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.125 int unpack_dms_UIMVerifyPIN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPINProtection_t * *pOutput*)

To get UIM verify PIN unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.4.2.126 int unpack_dms_ValidateSPC (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactoryDefaults_t * *pOutput*)

Requests the device to perform automatic service activation unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.5 fms.h File Reference

Data Structures

- struct CarrierImage_t
- struct pack_fms_GetImagesPreference_t
- struct FMSImageElement
- struct FMSPrefImageList
- struct unpack_fms_GetImagesPreference_t
- struct pack_fms_GetStoredImages_t
- struct FMSImageIdElement
- struct FMSImageIDEntries
- struct FMSImageList
- struct unpack_fms_GetStoredImages_t
- struct pack_fms_SetImagesPreference_t
- struct unpack_fms_SetImagesPreference_t

Macros

- #define FMS_GOBI_MBN_IMG_ID_STR_LEN 16
- #define FMS_GOBI_MBN_BUILD_ID_STR_LEN 100
- #define FMS_GOBI_LISTENTRIES_MAX 2
- #define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255
- #define FMS_MAX_IMAGE_ID_ELEMENT 50
- #define FMS_IMAGE_ID_MAX_ENTRIES 2
- #define FMS_FW_PRI_BUILD_MATCH_LEN 11
- #define FMS_IMAGE_ID_IMG_ID_LEN 16
- #define FMS_IMAGE_ID_BUILD_ID_LEN 32
- #define FMS_IMAGE_ID_PRI_IMGTYPE 0x01

Functions

- int pack_fms_GetImagesPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetImagesPreference_t *reqArg)
- int unpack_fms_GetImagesPreference (uint8_t *pResp, uint16_t respLen, unpack_fms_GetImagesPreference_t *pOutput)
- int pack_fms_GetStoredImages (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_GetStoredImages_t *reqArg)
- int unpack_fms_GetStoredImages (uint8_t *pResp, uint16_t respLen, unpack_fms_GetStoredImages_t *pOutput)
- int pack_fms_SetImagesPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_fms_SetImagesPreference_t *reqArg)
- int unpack_fms_SetImagesPreference (uint8_t *pResp, uint16_t respLen, unpack_fms_SetImagesPreference_t *pOutput)
- uint32_t GetValidFwPriCombinations (FMSImageList *pStoredImageList, uint32_t *pValidCombinationSize, CarrierImage_t *pValidCombinations)

9.5.1 Macro Definition Documentation

9.5.1.1 #define FMS_FW_PRI_BUILD_MATCH_LEN 11

9.5.1.2 #define FMS_GOBI_LISTENTRIES_MAX 2

9.5.1.3 #define FMS_GOBI_MBN_BUILD_ID_STR_LEN 100

9.5.1.4 #define FMS_GOBI_MBN_IMG_ID_STR_LEN 16

9.5.1.5 #define FMS_IMAGE_ID_BUILD_ID_LEN 32

9.5.1.6 #define FMS_IMAGE_ID_IMG_ID_LEN 16

9.5.1.7 #define FMS_IMAGE_ID_MAX_ENTRIES 2

9.5.1.8 #define FMS_IMAGE_ID_PRI_IMGTYPE 0x01

9.5.1.9 #define FMS_MAX_IMAGE_ID_ELEMENT 50

9.5.1.10 #define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255

9.5.2 Function Documentation

9.5.2.1 uint32_t GetValidFwPriCombinations (FMSImageList * *pStoredImageList*, uint32_t * *pValidCombinationSize*, CarrierImage_t * *pValidCombinations*)

This API distills valid Firmware/PRI combinations from GetStoredImages result

Parameters

in	*pStoredImage-List*	• image list returned from GetStoredImages • See FMSImageList
in,out	*pValid-CombinationSize*	• number of combination passed in and returned
out	*pValid-Combinations*	• valid combinations returned • See CarrierImage_t

Returns

- eQCWWAN_ERR_INVALID_ARG - Invalid parameters
- eQCWWAN_ERR_BUFFER_SZ - No enough element to store combinatons returned

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.5.2.2 int pack_fms_GetImagesPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_fms_GetImagesPreference_t * *reqArg*)

Get Images Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.5.2.3 int pack_fms_GetStoredImages (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_fms_GetStoredImages_t * *reqArg*)

Get Images Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.5.2.4 int pack_fms_SetImagesPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_fms_SetImagesPreference_t * *reqArg*)

Set Images Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.5.2.5 int unpack_fms_GetImagesPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_fms_GetImagesPreference_t * *pOutput*)

Get Images Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

9.5.2.6 int unpack_fms_GetStoredImages (uint8_t * *pResp*, uint16_t *respLen*, unpack_fms_GetStoredImages_t * *pOutput*)

Get Images Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

9.5.2.7 int unpack_fms_SetImagesPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_fms_SetImagesPreference_t * *pOutput*)

Set Images Preference unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

9.6 lite-fw.h File Reference

Data Structures

- struct _litefw_FirmwareInfo_

Macros

- #define IMG_MASK_MDM (1<<0)
- #define IMG_MASK_PRI (1<<1)
- #define IMG_MASK_CLEAR (0x000)
- #define IMG_MASK_GENERIC (0xFFFF)
- #define LITEFW_CARRIER_PACKAGE_SKU "9999999"
- #define LITEFW_SKU_STRING_LENGTH 7
- #define FIRMWARE_INFO_STRING_SIZE 20
- #define libSDP_GetVersion litefw_GetVersion
- #define libSDP_CalculateImageMask litefw_CalculateImageMask
- #define libSDP_getFileType litefw_getFileType
- #define libSDP_ExtractFirmwareParametersByPath litefw_ExtractFirmwareParametersByPath
- #define libSDP_GetModelFamily litefw_GetModelFamily
- #define libSDP_CheckValidFirmwareInfo litefw_CheckValidFirmwareInfo
- #define libSDP_BuildImagesPreferenceRequest litefw_BuildImagesPreferenceRequest
- #define libSDP_DownloadFW litefw_DownloadFW
- #define libsdp_set_log_func litefw_set_log_func
- #define libsdp_SetReadBlockSize litefw_SetReadBlockSize
- #define LIBSDP_CARRIER_PACKAGE_SKU LITEFW_CARRIER_PACKAGE_SKU
- #define LIBSDP_SKU_STRING_LENGTH LITEFW_SKU_STRING_LENGTH

Typedefs

- typedef struct _litefw_FirmwareInfo_ litefw_FirmwareInfo
- typedef void(* litefwlogger)(uint8_t lvl, const char *buff)
- typedef litefw_FirmwareInfo libSDP_FirmwareInfo
- typedef litefwlogger libsdplogger

Enumerations

- enum litefw_fwdwl_error_codes {
  eSDP_FWDWL_SUCCESS = 0,
  eSDP_FWDWL_ERR_GENERAL = 100,
  eSDP_FWDWL_ERR_SDK,
  eSDP_FWDWL_ERR_SET_CBK,
  eSDP_FWDWL_ERR_PATH_TOO_LONG,
  eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED,
  eSDP_FWDWL_ERR_FW_UPGRADE,
  eSDP_FWDWL_ERR_INVALID_DEV,
  eSDP_FWDWL_ERR_INVALID_PATH,
  eSDP_FWDWL_ERR_TIMEOUT,
  eSDP_FWDWL_ERR_FAIL,
  eSDP_FWDWL_ERR_PRI_FAIL,
  eSDP_FWDWL_ERR_FW_VERSION_FAIL,
  eSDP_FWDWL_ERR_END }
- enum litefw_Fw_Type {
  eFW_TYPE_MBN_GOBI,
  eFW_TYPE_MBN,
  eFW_TYPE_CWE,
  eFW_TYPE_NVU,
  eFW_TYPE_SPK,
  eFW_TYPE_INVALID,
  eFW_TYPE_CWE_NVU }
- enum litefw_Models {
  eModel_Unknown =-1,
  eModel_9X15 =0,
  eModel_WP9X15 =1,
  eModel_9X30 =2,
  eModel_9x07 =3 }

Functions

- int litefw_ExtractFirmwareParametersByPath (char *pImagePath, litefw_FirmwareInfo *info)
- int litefw_BuildImagesPreferenceRequest (litefw_FirmwareInfo info, pack_fms_SetImagesPreference_t *pack)
- int litefw_CalculateImageMask (unpack_fms_SetImagesPreference_t SetPrefRspFromModem)
- int litefw_getFileType (char *szPath)
- unsigned int litefw_DownloadFW (char *pImagePath, char *szTTYPath, int iFWImageType, int image_mask, int iModelFamily)
- int litefw_GetModelFamily (char *pModelString)
- int litefw_CheckValidFirmwareInfo (litefw_FirmwareInfo info)
- char * litefw_GetVersion ()
- int litefw_set_log_func (litefwlogger func)
- void litefw_SetReadBlockSize (unsigned long lBlockSize)
- int litefw_switch_9x07_to_downloadmode (char *szTTYPath)

9.6.1 Detailed Description

Filename: lite-fw.h

Purpose: Global definitions used inside the SDK

Copyright: © 2016 Sierra Wireless Inc., all rights reserved

9.6.2 Macro Definition Documentation

9.6.2.1 #define FIRMWARE_INFO_STRING_SIZE 20

9.6.2.2 #define IMG_MASK_CLEAR (0x000)

9.6.2.3 #define IMG_MASK_GENERIC (0xFFFF)

9.6.2.4 #define IMG_MASK_MDM (1<<0)

9.6.2.5 #define IMG_MASK_PRI (1<<1)

9.6.2.6 #define libSDP_BuildImagesPreferenceRequest litefw_BuildImagesPreferenceRequest

9.6.2.7 #define libSDP_CalculateImageMask litefw_CalculateImageMask

9.6.2.8 #define LIBSDP_CARRIER_PACKAGE_SKU LITEFW_CARRIER_PACKAGE_SKU

9.6.2.9 #define libSDP_CheckValidFirmwareInfo litefw_CheckValidFirmwareInfo

9.6.2.10 #define libSDP_DownloadFW litefw_DownloadFW

9.6.2.11 #define libSDP_ExtractFirmwareParametersByPath litefw_ExtractFirmwareParametersByPath

9.6.2.12 #define libSDP_getFileType litefw_getFileType

9.6.2.13 #define libSDP_GetModelFamily litefw_GetModelFamily

9.6.2.14 #define libSDP_GetVersion litefw_GetVersion

9.6.2.15 #define libsdp_set_log_func litefw_set_log_func

9.6.2.16 #define libsdp_SetReadBlockSize litefw_SetReadBlockSize

9.6.2.17 #define LIBSDP_SKU_STRING_LENGTH LITEFW_SKU_STRING_LENGTH

9.6.2.18 #define LITEFW_CARRIER_PACKAGE_SKU "9999999"

9.6.2.19 #define LITEFW_SKU_STRING_LENGTH 7

9.6.3 Typedef Documentation

9.6.3.1 typedef litefw_FirmwareInfo libSDP_FirmwareInfo

9.6.3.2 typedef litefwlogger libsdplogger

9.6.3.3 typedef struct _litefw_FirmwareInfo_ litefw_FirmwareInfo

This structure contains information of the response parameters associated with a Read Transparent API.

Parameters

szModelid_str	• Model Name String

szFwversion_str	• Firmware Version String.
szSku_str	• SKU String.
szPackageid_str	• Package ID String.
szCarrier_str	• Carrier String.
szCarrier-Priversion_str	• Carrier PRI Version String.

9.6.3.4 typedef void(∗ litefwlogger)(uint8_t lvl, const char ∗buff)

This Custom Log prototype.

Parameters

in	*lvl*	• Log level.
in	*buff*	• Log String.

Returns

none

9.6.4 Enumeration Type Documentation

9.6.4.1 enum litefw_Fw_Type

Enumerator

eFW_TYPE_MBN_GOBI
eFW_TYPE_MBN
eFW_TYPE_CWE
eFW_TYPE_NVU
eFW_TYPE_SPK
eFW_TYPE_INVALID
eFW_TYPE_CWE_NVU

9.6.4.2 enum litefw_fwdwl_error_codes

Enumerator

eSDP_FWDWL_SUCCESS

eSDP_FWDWL_ERR_GENERAL

eSDP_FWDWL_ERR_SDK

eSDP_FWDWL_ERR_SET_CBK

eSDP_FWDWL_ERR_PATH_TOO_LONG

eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED

eSDP_FWDWL_ERR_FW_UPGRADE

eSDP_FWDWL_ERR_INVALID_DEV

eSDP_FWDWL_ERR_INVALID_PATH

eSDP_FWDWL_ERR_TIMEOUT

eSDP_FWDWL_ERR_FAIL

eSDP_FWDWL_ERR_PRI_FAIL

eSDP_FWDWL_ERR_FW_VERSION_FAIL

eSDP_FWDWL_ERR_END

9.6.4.3 enum litefw_Models

Enumerator

eModel_Unknown

eModel_9X15

eModel_WP9X15

eModel_9X30

eModel_9x07

9.6.5 Function Documentation

9.6.5.1 int litefw_BuildImagesPreferenceRequest (litefw_FirmwareInfo *info*, pack_fms_SetImagesPreference_t * *pack*)

This API Build Image Preference Request Using Firmware Information.

Parameters

in	*info*	• See litefw_FirmwareInfo for more information.
out	*pack*	• See pack_fms_SetImagesPreference_t for more information.

Returns

0 on success, litefw_fwdwl_error_codes error value otherwise

See Also

See litefw_fwdwl_error_codes for error values

9.6.5.2 int litefw_CalculateImageMask (unpack_fms_SetImagesPreference_t *SetPrefRspFromModem*)

This API Calculate Image Mask for Firmware Download.

Parameters

in	*SetPrefRsp-FromModem*	• See unpack_fms_SetImagesPreference_t for more information.

Returns

Image Mask

- IMG_MASK_MDM | IMG_MASK_PRI

9.6.5.3 int litefw_CheckValidFirmwareInfo (litefw_FirmwareInfo *info*)

This API Check Valid Firmware Information to build Image Prefernce Request.

Parameters

in	*info*	• See litefw_FirmwareInfo for more information.

Returns

0 on success, -1 error value otherwise

9.6.5.4 unsigned int litefw_DownloadFW (char * *pImagePath*, char * *szTTYPath*, int *iFWImageType*, int *image_mask*, int *iModelFamily*)

This API Download Firmware.

Parameters

in	*pImagePath*	• Firmware Folder Path.
in	*szTTYPath*	• QDL Device Path.
in	*iFWImageType*	• Firmware Type. • See litefw_Fw_Type
in	*image_mask*	• Image Mask. – IMG_MASK_MDM | IMG_MASK_PRI
in	*iModelFamily*	• Modem Family. • See litefw_Models

Returns

0 on success, litefw_fwdwl_error_codes error value otherwise

See Also

See litefw_fwdwl_error_codes for error values

9.6.5.5 int litefw_ExtractFirmwareParametersByPath (char * *pImagePath,* litefw_FirmwareInfo * *info*)

This API Extrace Firmware Parameters From Path.

Parameters

in	*pImagePath*	• Firmware Folder Path.
out	*info*	• See litefw_FirmwareInfo for more information.

Returns

0 on success, litefw_fwdwl_error_codes error value otherwise

See Also

See litefw_fwdwl_error_codes for error values

9.6.5.6 int litefw_getFileType (char * *szPath*)

This API Get File Type By Path

Parameters

in	*szPath*	• See litefw_FirmwareInfo for more information.

Returns

eFW_TYPE_INVALID on error, litefw_Fw_Type value otherwise

See Also

See litefw_Fw_Type for values

9.6.5.7 int litefw_GetModelFamily (char * *pModelString*)

This API Get Model Famliy from a model string.

Parameters

in	*pModelString*	• Model String.

Returns

eModel_Unknown on Error, litefw_Models value otherwise

See Also

See litefw_Models for values

9.6.5.8 char* litefw_GetVersion ( )

This API Get Lib SDP Version.

Returns

Version String

9.6.5.9 int litefw_set_log_func (litefwlogger *func*)

This API Set Custom Log function.

Parameters

in	*func*	• See litefwlogger for more information.

Returns

none

9.6.5.10 void litefw_SetReadBlockSize (unsigned long *lBlockSize*)

This API Set Read BlockSize.

Parameters

in	*lBlockSize*	• Firmware Read Block Size.

Returns

none

9.6.5.11 int litefw_switch_9x07_to_downloadmode (char * *szTTYPath*)

This API switch 9x07 modem to download mode.

Parameters

in	*szTTYPath*	• QDL Device Path.

Returns

0 on success, -1 error value otherwise

9.7 loc.h File Reference

Data Structures

- struct loc_LocApplicationInfo
- struct loc_SV
- struct loc_SVInfo
- struct loc_GnssData
- struct loc_CellDb
- struct loc_ClkInfo
- struct loc_BdsSV
- struct loc_BdsSVInfo
- struct pack_loc_EventRegister_t
- struct unpack_loc_EventRegister_t
- struct pack_loc_SetExtPowerState_t
- struct unpack_loc_SetExtPowerState_t
- struct pack_loc_Start_t
- struct unpack_loc_Start_t
- struct pack_loc_Stop_t
- struct unpack_loc_Stop_t
- struct pack_loc_SetOperationMode_t
- struct unpack_loc_SetOperationMode_t
- struct pack_loc_Delete_Assist_Data_t
- struct unpack_loc_Delete_Assist_Data_t
- struct loc_precisionDilution
- struct loc_sensorDataUsage
- struct loc_svUsedforFix
- struct loc_gpsTime
- struct unpack_loc_PositionRpt_Ind_t
- struct unpack_loc_EngineState_Ind_t
- struct unpack_loc_SetExtPowerConfig_Ind_t
- struct unpack_loc_SLQSLOCGetBestAvailPos_t
- struct pack_loc_SLQSLOCGetBestAvailPos_t
- struct unpack_loc_BestAvailPos_Ind_t
- struct unpack_loc_SetOperationMode_Ind_t
- struct unpack_loc_DeleteAssistData_Ind_t
- struct loc_satelliteInfo
- struct unpack_loc_GnssSvInfo_Ind_t
- struct pack_loc_SLQSLOCInjectUTCTime_t
- struct altSrcInfo_t
- struct pack_loc_SLQSLOCInjectPosition_t
- struct pack_loc_SLQSLOCSetCradleMountConfig_t
- struct sensorData_t
- struct tempData_t
- struct pack_loc_SLQSLOCInjectSensorData_t
- struct unpack_loc_EventNMEA_Ind_t
- struct pack_loc_SLQSLOCGetServer_t
- struct loc_IPv4Info
- struct loc_IPv6Info
- struct loc_urlAddr
- struct unpack_loc_GetServer_Ind_t

Macros

- #define LOC_UINT8_MAX_STRING_SZ 255
- #define LOCEVENTMASKPOSITIONREPORT 0x00000001
- #define LOCEVENTMASKGNSSSVINFO 0x00000002
- #define LOCEVENTMASKNMEA 0x00000004
- #define LOCEVENTMASKNINOTIFYVERIFYREQ 0x00000008
- #define LOCEVENTMASKINJECTTIMEREQ 0x00000010
- #define LOCEVENTMASKINJECTPREDICTEDORBITSREQ 0x00000020
- #define LOCEVENTMASKINJECTPOSITIONREQ 0x00000040
- #define LOCEVENTMASKENGINESTATE 0x00000080
- #define LOCEVENTMASKFIXSESSIONSTATE 0x00000100
- #define LOCEVENTMASKWIFIREQ 0x00000200
- #define LOCEVENTMASKSENSORSTREAMINGREADYSTATUS 0x00000400
- #define LOCEVENTMASKTIMESYNCREQ 0x00000800
- #define LOCEVENTMASKSETSPISTREAMINGREPORT 0x00001000
- #define LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ 0x00002000
- #define LOCEVENTMASKNIGEOFENCENOTIFICATION 0x00004000
- #define LOCEVENTMASKGEOFENCEGENALERT 0x00008000
- #define LOCEVENTMASKGEOFENCEBREACHNOTIFICATION 0x00010000
- #define LOCEVENTMASKPEDOMETERCONTROL 0x00020000
- #define LOCEVENTMASKMOTIONDATACONTROL 0x00040000
- #define LOCEVENTMASKBATCHFULLNOTIFICATION 0x00080000
- #define LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT 0x00100000
- #define LOCEVENTMASKINJECTWIFIAPDATAREQ 0x00200000
- #define LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION 0x00400000
- #define LOCEVENTMASKVEHICLEDATAREADYSTATUS 0x00800000
- #define LOCEVENTMASKGNSSMEASUREMENTREPORT 0x01000000
- #define LOCEVENTMASKINVALIDVALUE 0xFFFFFFFF
- #define MAX_SENSOR_DATA_LEN 64
- #define MAX_TEMP_DATA_LEN 64
- #define MAX_LOC_NMEA_STR_LEN 201

Enumerations

- enum {
  eQMI_LOC_SESS_STATUS_SUCCESS =0,
  eQMI_LOC_SESS_STATUS_IN_PROGRESS =1,
  eQMI_LOC_SESS_STATUS_FAILURE =2,
  eQMI_LOC_SESS_STATUS_TIMEOUT =3 }

Functions

- int pack_loc_EventRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_EventRegister_t *reqArg)
- int unpack_loc_EventRegister (uint8_t *pResp, uint16_t respLen, unpack_loc_EventRegister_t *pOutput)
- int pack_loc_SetExtPowerState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SetExtPowerState_t *reqArg)
- int unpack_loc_SetExtPowerState (uint8_t *pResp, uint16_t respLen, unpack_loc_SetExtPowerState_t *pOutput)
- int pack_loc_Start (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Start_t *reqArg)
- int unpack_loc_Start (uint8_t *pResp, uint16_t respLen, unpack_loc_Start_t *pOutput)
- int pack_loc_Stop (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Stop_t *reqArg)
- int unpack_loc_Stop (uint8_t *pResp, uint16_t respLen, unpack_loc_Stop_t *pOutput)

- int pack_loc_SetOperationMode (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SetOperationMode_t *reqArg)
- int unpack_loc_SetOperationMode (uint8_t *pResp, uint16_t respLen, unpack_loc_SetOperationMode_t *pOutput)
- int pack_loc_DeleteAssistData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_Delete_Assist_Data_t *reqArg)
- int unpack_loc_DeleteAssistData (uint8_t *pResp, uint16_t respLen, unpack_loc_Delete_Assist_Data_t *pOutput)
- int unpack_loc_PositionRpt_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_PositionRpt_Ind_t *pOutput)
- int unpack_loc_EngineState_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_EngineState_Ind_t *pOutput)
- int unpack_loc_SetExtPowerConfig_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_SetExtPowerConfig_Ind_t *pOutput)
- int pack_loc_SLQSLOCGetBestAvailPos (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCGetBestAvailPos_t *reqArg)
- int unpack_loc_SLQSLOCGetBestAvailPos (uint8_t *pResp, uint16_t respLen, unpack_loc_SLQSLOCGetBestAvailPos_t *pOutput)
- int unpack_loc_BestAvailPos_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_BestAvailPos_Ind_t *pOutput)
- int unpack_loc_SetOperationMode_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_SetOperationMode_Ind_t *pOutput)
- int unpack_loc_DeleteAssistData_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_DeleteAssistData_Ind_t *pOutput)
- int unpack_loc_GnssSvInfo_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_GnssSvInfo_Ind_t *pOutput)
- int pack_loc_SLQSLOCInjectUTCTime (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectUTCTime_t *reqArg)
- int unpack_loc_SLQSLOCInjectUTCTime (uint8_t *pResp, uint16_t respLen)
- int pack_loc_SLQSLOCInjectPosition (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectPosition_t *reqArg)
- int unpack_loc_SLQSLOCInjectPosition (uint8_t *pResp, uint16_t respLen)
- int pack_loc_SLQSLOCSetCradleMountConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCSetCradleMountConfig_t *reqArg)
- int unpack_loc_SLQSLOCSetCradleMountConfig (uint8_t *pResp, uint16_t respLen)
- int pack_loc_SLQSLOCInjectSensorData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCInjectSensorData_t *reqArg)
- int unpack_loc_SLQSLOCInjectSensorData (uint8_t *pResp, uint16_t respLen)
- int unpack_loc_EventNMEA_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_EventNMEA_Ind_t *pOutput)
- int pack_loc_SLQSLOCGetServer (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_loc_SLQSLOCGetServer_t *reqArg)
- int unpack_loc_SLQSLOCGetServer (uint8_t *pResp, uint16_t respLen)
- int unpack_loc_GetServer_Ind (uint8_t *pResp, uint16_t respLen, unpack_loc_GetServer_Ind_t *pOutput)

9.7.1 Macro Definition Documentation

9.7.1.1 #define LOC_UINT8_MAX_STRING_SZ 255

9.7.1.2 #define LOCEVENTMASKBATCHFULLNOTIFICATION 0x00080000

The control point must enable this mask to receive notification when a batch is full. The location engine sends this event to notify of Batch Full for ongoing batching session.

9.7.1.3 #define LOCEVENTMASKENGINESTATE 0x00000080

The control point must enable this mask to receive engine state report event indications.

9.7.1.4 #define LOCEVENTMASKFIXSESSIONSTATE 0x00000100

The control point must enable this mask to receive fix session status report event indications.

9.7.1.5 #define LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION 0x00400000

The control point must enable this mask to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach notification is for multiple Geofences. Breaches from multiple Geofences are all batched and sent in the same notification.

9.7.1.6 #define LOCEVENTMASKGEOFENCEBREACHNOTIFICATION 0x00010000

The control point must enable this mask to receive notifications when a Geofence is breached. These events are generated when a UE enters or leaves the perimeter of a Geofence. This breach report is for a single Geofence.

9.7.1.7 #define LOCEVENTMASKGEOFENCEGENALERT 0x00008000

The control point must enable this mask to receive Geofence alerts. These alerts are generated to inform the client of the changes that may affect a Geofence, for example, if GPS is turned off or if the network is unavailable.

9.7.1.8 #define LOCEVENTMASKGNSSMEASUREMENTREPORT 0x01000000

The control point must enable this mask to receive system clock and satellite measurement report events (system clock, SV time, Doppler, etc.). Reports are generated only for the GNSS satellite constellations that are enabled using QMI_LOC_SET_GNSS_ CONSTELL_REPORT_CONFIG(Not yet supported).

9.7.1.9 #define LOCEVENTMASKGNSSSVINFO 0x00000002

The control point must enable this mask to receive satellite report event indications. These reports are sent at a 1 Hz rate.

9.7.1.10 #define LOCEVENTMASKINJECTPOSITIONREQ 0x00000040

The control point must enable this mask to receive position injection request event indications.

9.7.1.11 #define LOCEVENTMASKINJECTPREDICTEDORBITSREQ 0x00000020

The control point must enable this mask to receive predicted orbits request event indications.

9.7.1.12 #define LOCEVENTMASKINJECTTIMEREQ 0x00000010

The control point must enable this mask to receive time injection request event indications.

9.7.1.13 #define LOCEVENTMASKINJECTWIFIAPDATAREQ 0x00200000

The control point must enable this mask to receive Wi-Fi Access Point (AP) data inject request event indications.

9.7.1.14 #define LOCEVENTMASKINVALIDVALUE 0xFFFFFFFF

Invalid Event Mask

9.7.1.15 #define LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT 0x00100000

The control point must enable this mask to receive position report indications along with an ongoing batching session. The location engine sends this event to notify the batched position report while a batching session is ongoing.

9.7.1.16 #define LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ 0x00002000

The control point must enable this mask to receive location server requests. These requests are generated when the service wishes to establish a connection with a location server.

9.7.1.17 #define LOCEVENTMASKMOTIONDATACONTROL 0x00040000

The control point must enable this mask to register for motion data control requests from the location engine. The location engine sends this event to control the injection of motion data.

9.7.1.18 #define LOCEVENTMASKNIGEOFENCENOTIFICATION 0x00004000

The control point must enable this mask to receive notifications related to network-initiated Geofences. These events notify the client when a network-initiated Geofence is added, deleted, or edited.

9.7.1.19 #define LOCEVENTMASKNINOTIFYVERIFYREQ 0x00000008

The control point must enable this mask to receive NI Notify/Verify request event indications.

9.7.1.20 #define LOCEVENTMASKNMEA 0x00000004

The control point must enable this mask to receive NMEA reports for position and satellites in view. The report is at a 1 Hz rate.

9.7.1.21 #define LOCEVENTMASKPEDOMETERCONTROL 0x00020000

The control point must enable this mask to register for pedometer control requests from the location engine. The location engine sends this event to control the injection of pedometer reports.

9.7.1.22 #define LOCEVENTMASKPOSITIONREPORT 0x00000001

The control point must enable this mask to receive position report event indications.

9.7.1.23 #define LOCEVENTMASKSENSORSTREAMINGREADYSTATUS 0x00000400

The control point must enable this mask to receive notifications from the location engine indicating its readiness to accept data from the sensors (accelerometer, gyroscope, etc.).

9.7.1.24 #define LOCEVENTMASKSETSPISTREAMINGREPORT 0x00001000

The control point must enable this mask to receive Stationary Position Indicator (SPI) streaming report indications.

9.7.1.25 #define LOCEVENTMASKTIMESYNCREQ 0x00000800

The control point must enable this mask to receive time sync requests from the GPS engine. Time sync enables the GPS engine to synchronize its clock with the sensor processor's clock.

9.7.1.26 #define LOCEVENTMASKVEHICLEDATAREADYSTATUS 0x00800000

The control point must enable this mask to receive notifications from the location engine indicating its readiness to accept vehicle data (vehicle accelerometer, vehicle angular rate, vehicle odometry, etc.).

9.7.1.27 #define LOCEVENTMASKWIFIREQ 0x00000200

The control point must enable this mask to receive Wi-Fi position request event indications.

9.7.1.28 #define MAX_LOC_NMEA_STR_LEN 201

9.7.1.29 #define MAX_SENSOR_DATA_LEN 64

9.7.1.30 #define MAX_TEMP_DATA_LEN 64

9.7.2 Enumeration Type Documentation

9.7.2.1 anonymous enum

Enumerator

eQMI_LOC_SESS_STATUS_SUCCESS

eQMI_LOC_SESS_STATUS_IN_PROGRESS

eQMI_LOC_SESS_STATUS_FAILURE

eQMI_LOC_SESS_STATUS_TIMEOUT

9.7.3 Function Documentation

9.7.3.1 int pack_loc_DeleteAssistData (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* pack_loc_Delete_Assist_Data_t * *reqArg*)

Delete Assistant Data pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.2 int pack_loc_EventRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_EventRegister_t * *reqArg*)

Event Register pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.3 int pack_loc_SetExtPowerState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SetExtPowerState_t * *reqArg*)

Set Ext Power State pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.4 int pack_loc_SetOperationMode (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SetOperationMode_t * *reqArg*)

Set Operation Mode pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.5 int pack_loc_SLQSLOCGetBestAvailPos (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCGetBestAvailPos_t * *reqArg*)

Get Best Avail position pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.6 int pack_loc_SLQSLOCGetServer (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCGetServer_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

The request is acknowledged through the response, but the SUCCESS/FAILURE status is sent through unpack_loc_GetServer_Ind callback If successful, the callback also contains the A-GPS server address.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.7 int pack_loc_SLQSLOCInjectPosition (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCInjectPosition_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.8 int pack_loc_SLQSLOCInjectSensorData (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCInjectSensorData_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.9 int pack_loc_SLQSLOCInjectUTCTime (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCInjectUTCTime_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.10 int pack_loc_SLQSLOCSetCradleMountConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_SLQSLOCSetCradleMountConfig_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.11 int pack_loc_Start (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_Start_t * *reqArg*)

LOC Start pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.12 int pack_loc_Stop (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_Stop_t * *reqArg*)

Loc Stop pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.13 int unpack_loc_BestAvailPos_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_BestAvailPos_Ind_t * *pOutput*)

Loc Best Avial position Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.14 int unpack_loc_DeleteAssistData (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Delete_Assist_Data_t * *pOutput*)

Delete Assistant Data unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.15 int unpack_loc_DeleteAssistData_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_DeleteAssistData_Ind_t * *pOutput*)

Unpack the status of delete the location engine assistance data

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.16 int unpack_loc_EngineState_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EngineState_Ind_t * *pOutput*)

Loc Engine State Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.17 int unpack_loc_EventNMEA_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EventNMEA_Ind_t * *pOutput*)

Unpack the NMEA Event Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.18 int unpack_loc_EventRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EventRegister_t * *pOutput*)

Event Register unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.19 int unpack_loc_GetServer_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_GetServer_Ind_t * *pOutput*)

Unpack the Get Server Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.20 int unpack_loc_GnssSvInfo_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_GnssSvInfo_Ind_t * *pOutput*)

Unpack the GNSS SV Info Indication.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.21 int unpack_loc_PositionRpt_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_PositionRpt_Ind_t * *pOutput*)

Loc Position Report Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.22 int unpack_loc_SetExtPowerConfig_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetExtPowerConfig_Ind_t * *pOutput*)

Loc Set External Power Configure Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.23 int unpack_loc_SetExtPowerState (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetExtPowerState_t * *pOutput*)

Set Ext Power State unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.24 int unpack_loc_SetOperationMode (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetOperationMode_t * *pOutput*)

Set Operation Mode unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.25 int unpack_loc_SetOperationMode_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SetOperationMode_Ind_t * *pOutput*)

Unpack the engine to use the specified operation mode.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.26 int unpack_loc_SLQSLOCGetBestAvailPos (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_SLQSLOCGetBestAvailPos_t * *pOutput*)

Get Best Avail position unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.27 int unpack_loc_SLQSLOCGetServer (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.28 int unpack_loc_SLQSLOCInjectPosition (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.29 int unpack_loc_SLQSLOCInjectSensorData (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.30 int unpack_loc_SLQSLOCInjectUTCTime (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.31 int unpack_loc_SLQSLOCSetCradleMountConfig (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.32 int unpack_loc_Start (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Start_t * *pOutput*)

Loc Start unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.33 int unpack_loc_Stop (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Stop_t * *pOutput*)

Loc Stop unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8 nas.h File Reference

Data Structures

- struct unpack_nas_GetSignalStrengths_t
- struct unpack_nas_SLQSGetSysSelectionPref_t
- struct nas_netSelectionPref
- struct nas_acqOrderPref
- struct nas_CSGID
- struct pack_nas_SLQSSetSysSelectionPref_t
- struct pack_nas_SLQSSetBandPreference_t
- struct pack_nas_SLQSNasIndicationRegisterExt_t
- struct RFBandInfoElements
- struct unpack_nas_GetRFInfo_t
- struct cdmaSSInfo
- struct hdrSSInfo
- struct lteSSInfo
- struct tdscdmaSigInfoExt
- struct unpack_nas_SLQSNasGetSigInfo_t
- struct unpack_nas_SLQSNasSigInfoCallback_ind_t
- struct unpack_nas_GetHomeNetwork_t
- struct nas_SrvStatusInfo
- struct nas_GSMSrvStatusInfo
- struct nas_sysInfoCommon
- struct nas_CDMASysInfo
- struct nas_HDRSysInfo
- struct nas_GSMSysInfo
- struct nas_WCDMASysInfo
- struct nas_LTESysInfo
- struct nas_AddCDMASysInfo
- struct nas_AddSysInfo
- struct nas_CallBarringSysInfo
- struct unpack_nas_SLQSGetSysInfo_t
- struct unpack_nas_SLQSSysInfoCallback_ind_t
- struct unpack_nas_GetServingNetwork_t
- struct unpack_nas_GetServingNetworkCapabilities_t
- struct nas_QmiNas3GppNetworkInfo
- struct nas_QmiNas3GppNetworkRAT
- struct nas_QmisNasPcsDigit
- struct unpack_nas_PerformNetworkScan_t
- struct unpack_nas_SLQSSwiGetLteCQI_t
- struct nas_CommInfo
- struct nas_LTEInfo
- struct unpack_nas_SLQSNasSwiModemStatus_t
- struct nas_servSystem
- struct nas_dataSrvCapabilities
- struct nas_currentPLMN
- struct nas_roamIndList
- struct nas_qaQmi3Gpp2TimeZone
- struct nas_detailSvcInfo
- struct nas_CDMASysInfoExt
- struct nas_callBarStatus
- struct unpack_nas_SLQSGetServingSystem_t
- struct nas_rxSignalStrengthListElement
- struct nas_ecioListElement
- struct nas_errorRateListElement
- struct nas_rsrqInformation

- struct nas_lteSnrinformation
- struct nas_lteRsrpinformation
- struct unpack_nas_SLQSGetSignalStrength_t
- struct nas_SLQSSignalStrengthsIndReq
- struct pack_nas_SLQSSetSignalStrengthsCallback_t
- struct nas_SLQSSignalStrengthsInformation
- struct nas_RejectReasonTlv
- struct nas_SignalStrengthTlv
- struct nas_RFInfoTlv
- struct nas_SLQSSignalStrengthsTlv
- struct unpack_nas_SetEventReportInd_t
- struct unpack_nas_GetCDMANetworkParameters_t
- struct pack_nas_SetACCOLC_t
- struct nas_CDMARSSIThresh
- struct nas_CDMAECIOThresh
- struct nas_HDRRSSIThresh
- struct nas_HDRECIOThresh
- struct nas_HDRSINRThreshold
- struct nas_HDRIOThresh
- struct nas_GSMRSSIThresh
- struct nas_WCDMARSSIThresh
- struct nas_WCDMAECIOThresh
- struct nas_LTERSSIThresh
- struct nas_LTESNRThreshold
- struct nas_LTERSRQThresh
- struct nas_LTERSRPThresh
- struct nas_LTESigRptConfig
- struct nas_TDSCDMARSCPThresh
- struct nas_TDSCDMARSSIThresh
- struct nas_TDSCDMAECIOThresh
- struct nas_TDSCDMASINRThresh
- struct pack_nas_SLQSNasConfigSigInfo2_t
- struct unpack_nas_SetDataCapabilitiesCallback_ind_t
- struct unpack_nas_GetNetworkPreference_t
- struct pack_nas_SetNetworkPreference_t
- struct unpack_nas_SetNetworkPreference_t
- struct unpack_nas_SetRoamingIndicatorCallback_ind_t
- struct NASServingSystemInfo
- struct unpack_nas_SetServingSystemCallback_ind_t
- struct NASPhyCaAggScellIndType
- struct NASPhyCaAggScellDlBw
- struct NASPhyCaAggScellInfo
- struct NASPhyCaAggPcellInfo
- struct NASPhyCaAggScellIndex
- struct NasGetLTECphyCaInfo
- struct unpack_nas_SlqsGetLTECphyCAInfo_t
- struct NASEmergencyModeTlv
- struct NASModePreferenceTlv
- struct NASBandPreferenceTlv
- struct NASPRLPreferenceTlv
- struct NASRoamPreferenceTlv
- struct NASLTEBandPreferenceTlv
- struct NASNetSelPreferenceTlv
- struct NASServDomainPrefTlv
- struct NASGWAcqOrderPrefTlv

- struct NASQmiCbkNasSystemSelPrefInd
- struct unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t
- struct NASOTAMessageTlv
- struct NASLteNasReleaseInfoTlv
- struct NASTimeInfoTlv
- struct NASQmiCbkNasSwiOTAMessageInd
- struct unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t
- struct nas_MNRInfo
- struct pack_nas_SLQSInitiateNetworkRegistration_t
- struct pack_nas_SLQSNasSwiIndicationRegister_t
- struct pack_nas_SLQSGetPLMNName_t
- struct unpack_nas_SLQSGetPLMNName_t
- struct nas_nmrCellInfo
- struct nas_GERANInfo
- struct nas_geranInstInfo
- struct nas_UMTSinstInfo
- struct nas_UMTSInfo
- struct nas_CDMAInfo
- struct nas_cellParams
- struct nas_LTEInfoIntrafreq
- struct nas_infoInterFreq
- struct nas_LTEInfoInterfreq
- struct nas_gsmCellInfo
- struct nas_lteGsmCellInfo
- struct nas_LTEInfoNeighboringGSM
- struct nas_wcdmaCellInfo
- struct nas_lteWcdmaCellInfo
- struct nas_LTEInfoNeighboringWCDMA
- struct nas_umtsLTENbrCell
- struct nas_WCDMAInfoLTENeighborCell
- struct unpack_nas_SLQSNasGetCellLocationInfo_t
- struct nas_timeInfo
- struct unpack_nas_SLQSGetNetworkTime_t
- struct nas_UniversalTime
- struct unpack_nas_SLQSNasNetworkTimeCallBack_ind_t
- struct nas_PhyCaAggScellIndType
- struct nas_PhyCaAggScellDlBw
- struct nas_PhyCaAggScellInfo
- struct nas_PhyCaAggPcellInfo
- struct nas_PhyCaAggScellIndex
- struct unpack_nas_SetNasLTECphyCaIndCallback_ind_t
- struct nas_RxSigInfo
- struct nas_SccRxInfo
- struct unpack_nas_SLQSSwiGetLteSccRxInfo_t
- struct unpack_nas_SLQSNasTimerCallback_ind_t
- struct pack_nas_InitiateDomainAttach_t
- struct pack_nas_SetCDMANetworkParameters_t
- struct unpack_nas_SLQSNasGetHDRColorCode_t
- struct pack_nas_SLQSNasGetTxRxInfo_t
- struct nas_rxInfo
- struct nas_txInfo
- struct unpack_nas_SLQSNasGetTxRxInfo_t
- struct nas_OperatorPLMNData
- struct nas_operatorPLMNList
- struct nas_serviceProviderName

- struct nas_PLMNNetworkNameData
- struct nas_PLMNNetworkName
- struct nas_operatorNameString
- struct unpack_nas_SLQSGetOperatorNameData_t
- struct pack_nas_SLQSNasGet3GPP2Subscription_t
- struct nas_namName
- struct nas_dirNum
- struct nas_sidNid
- struct nas_homeSIDNID
- struct nas_minBasedIMSI
- struct nas_trueIMSI
- struct nas_CDMAChannel
- struct unpack_nas_SLQSNasGet3GPP2Subscription_t
- struct nas_protocolSubtypeElement
- struct unpack_nas_SLQSSwiGetHDRPersonality_t
- struct unpack_nas_SLQSSwiGetHDRProtSubtype_t
- struct pack_nas_SLQSSwiPSDetach_t
- struct unpack_nas_SLQSGetErrorRate_t
- struct nas_DRCParams
- struct nas_PilotSetParams
- struct nas_PilotSetData
- struct unpack_nas_SLQSSwiGetHRPDStats_t
- struct nas_ActPilotPNElement
- struct nas_NetworkStat1x
- struct nas_NetworkStatEVDO
- struct nas_DeviceConfigDetail
- struct nas_DataStatusDetail
- struct unpack_nas_SLQSSwiNetworkDebug_t
- struct nas_RSSIThresh
- struct nas_ECIOThresh
- struct nas_HDRSINRThresh
- struct nas_LTESNRThresh
- struct nas_IOThresh
- struct nas_RSRQThresh
- struct nas_RSRPThresh
- struct nas_LTESigRptCfg
- struct nas_TDSCDMASINRCONFThresh
- struct pack_nas_SLQSConfigSigInfo_t
- struct unpack_nas_GetHomeNetwork3GPP2_t
- struct nas_wcdmaUARFCN
- struct nas_lteEARFCN
- struct nas_ltePCI
- struct pack_nas_SLQSNASSwiSetChannelLock_t
- struct unpack_nas_SLQSNASSwiGetChannelLock_t

Macros

- #define NAS_OTA_MESSAGE_MAX_BUF_SIZE 2048
- #define NAS_MAX_NUM_NETWORKS 30
- #define NAS_MAX_DESCRIPTION_LENGTH 255
- #define NAS_PLMN_LENGTH 3
- #define NAS_MAX_SCC_RX_INFO_INSTANCES 255
- #define NAS_NAM_NAME_LENGTH 12
- #define NAS_IMSI_M_S1_LENGTH 7
- #define NAS_IMSI_M_S2_LENGTH 3
- #define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST 255

Enumerations

- enum LITEQMI_NAS_LTE_CPHY_SCELL_STATE {
  eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED =0x00,
  eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED =0x01,
  eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED =0x02 }
- enum LITEQMI_NAS_LTE_CPHY_CA_BW_NRB {
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6 =0x00,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15 =0x01,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25 =0x02,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50 =0x03,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75 =0x04,
  eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100 =0x05 }
- enum NAS_LTE_CPHY_CA_BW_NRB_LITE {
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_6 =0x00,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_15 =0x01,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_25 =0x02,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_50 =0x03,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_75 =0x04,
  eNAS_LTE_CPHY_CA_BW_NRB_LITE_100 =0x05 }
- enum NAS_LTE_CPHY_SCELL_STATE_LITE {
  eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE =0x00,
  eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE =0x01,
  eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE =0x02 }

Functions

- int unpack_nas_GetSignalStrengths (uint8_t *pResp, uint16_t respLen, unpack_nas_GetSignalStrengths_t *pOutput)
- int pack_nas_GetSignalStrengths (pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen)
- int pack_nas_SLQSGetSysSelectionPref (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetSysSelectionPref (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysSelectionPref_t *pOutput)
- int pack_nas_SLQSSetSysSelectionPref (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSysSelectionPref_t *pReqParam)
- int unpack_nas_SLQSSetSysSelectionPref (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSSetBandPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetBandPreference_t *pReqParam)
- int unpack_nas_SLQSSetBandPreference (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSNasIndicationRegisterExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasIndicationRegisterExt_t *pReqParam)
- int unpack_nas_SLQSNasIndicationRegisterExt (uint8_t *pResp, uint16_t respLen)
- int pack_nas_GetRFInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetRFInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_GetRFInfo_t *pOutput)
- int pack_nas_SLQSNasGetSigInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetSigInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetSigInfo_t *pOutput)
- int unpack_nas_SLQSNasSigInfoCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSigInfoCallback_ind_t *pOutput)
- int unpack_nas_GetHomeNetwork (uint8_t *pResp, uint16_t respLen, unpack_nas_GetHomeNetwork_t *pOutput)
- int pack_nas_GetHomeNetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int pack_nas_SLQSGetSysInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetSysInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSysInfo_t *pOutput)

- int unpack_nas_SLQSNasSysInfoCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSysInfoCallback_ind_t *pOutput)
- int pack_nas_GetServingNetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetServingNetwork (uint8_t *pResp, uint16_t respLen, unpack_nas_GetServingNetwork_t *pOutput)
- int pack_nas_GetServingNetworkCapabilities (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetServingNetworkCapabilities (uint8_t *pResp, uint16_t respLen, unpack_nas_GetServingNetworkCapabilities_t *pOutput)
- int pack_nas_PerformNetworkScan (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_PerformNetworkScan (uint8_t *pResp, uint16_t respLen, unpack_nas_PerformNetworkScan_t *pOutput)
- int pack_nas_SLQSSwiGetLteCQI (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetLteCQI (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteCQI_t *pOutput)
- int pack_nas_SLQSNasSwiModemStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasSwiModemStatus (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSwiModemStatus_t *pOutput)
- int pack_nas_SLQSGetServingSystem (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetServingSystem (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetServingSystem_t *pOutput)
- int pack_nas_SLQSGetSignalStrength (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint16_t reqMask)
- int unpack_nas_SLQSGetSignalStrength (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetSignalStrength_t *pOutput)
- int pack_nas_SLQSSetSignalStrengthsCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSetSignalStrengthsCallback_t *pReqParam)
- int unpack_nas_SLQSSetSignalStrengthsCallback (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SetRFInfoCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)
- int unpack_nas_SetRFInfoCallback (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SetLURejectCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, uint8_t *pBenable)
- int unpack_nas_SetLURejectCallback (uint8_t *pResp, uint16_t respLen)
- int unpack_nas_SetEventReportInd (uint8_t *pResp, uint16_t respLen, unpack_nas_SetEventReportInd_t *pOutput)
- int pack_nas_GetCDMANetworkParameters (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetCDMANetworkParameters (uint8_t *pResp, uint16_t respLen, unpack_nas_GetCDMANetworkParameters_t *pOutput)
- int pack_nas_GetANAAAAuthenticationStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetANAAAAuthenticationStatus (uint8_t *pResp, uint16_t respLen, uint32_t *pAuthStatus)
- int pack_nas_GetACCOLC (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetACCOLC (uint8_t *pResp, uint16_t respLen, uint8_t *pAccolc)
- int pack_nas_SetACCOLC (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetACCOLC_t reqParam)
- int unpack_nas_SetACCOLC (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSNasConfigSigInfo2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasConfigSigInfo2_t *pReqParam)
- int unpack_nas_SLQSNasConfigSigInfo2 (uint8_t *pResp, uint16_t respLen)
- int unpack_nas_SetDataCapabilitiesCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetDataCapabilitiesCallback_ind_t *pOutput)
- int pack_nas_GetNetworkPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetNetworkPreference (uint8_t *pResp, uint16_t respLen, unpack_nas_GetNetworkPreference_t *pOutput)
- int pack_nas_SetNetworkPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetNetworkPreference_t *reqArg)
- int unpack_nas_SetNetworkPreference (uint8_t *pResp, uint16_t respLen, unpack_nas_SetNetworkPreference_t *pOutput)

- int unpack_nas_SetRoamingIndicatorCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetRoamingIndicatorCallback_ind_t *pOutput)
- int unpack_nas_SetServingSystemCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetServingSystemCallback_ind_t *pOutput)
- int pack_nas_SlqsGetLTECphyCAInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SlqsGetLTECphyCAInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SlqsGetLTECphyCAInfo_t *pOutput)
- int unpack_nas_SLQSSetSysSelectionPrefCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t *pOutput)
- int unpack_nas_SLQSNasSwiOTAMessageCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t *pOutput)
- int pack_nas_SLQSInitiateNetworkRegistration (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSInitiateNetworkRegistration_t *pReqParam)
- int unpack_nas_SLQSInitiateNetworkRegistration (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSNasSwiIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasSwiIndicationRegister_t *pReqParam)
- int unpack_nas_SLQSNasSwiIndicationRegister (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSGetPLMNName (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSGetPLMNName_t *reqArg)
- int unpack_nas_SLQSGetPLMNName (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetPLMNName_t *pOutput)
- int pack_nas_SLQSNasGetCellLocationInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetCellLocationInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetCellLocationInfo_t *pOutput)
- int pack_nas_SLQSGetNetworkTime (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetNetworkTime (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetNetworkTime_t *pOutput)
- int unpack_nas_SLQSNasNetworkTimeCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasNetworkTimeCallBack_ind_t *pOutput)
- int unpack_nas_SetNasLTECphyCaIndCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SetNasLTECphyCaIndCallback_ind_t *pOutput)
- int pack_nas_SLQSSwiGetLteSccRxInfo (pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetLteSccRxInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetLteSccRxInfo_t *pOutput)
- int unpack_nas_SLQSNasTimerCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasTimerCallback_ind_t *pOutput)
- int pack_nas_InitiateDomainAttach (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_InitiateDomainAttach_t *pReqParam)
- int unpack_nas_InitiateDomainAttach (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SetCDMANetworkParameters (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SetCDMANetworkParameters_t *pReqParam)
- int unpack_nas_SetCDMANetworkParameters (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSNasGetHDRColorCode (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNasGetHDRColorCode (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetHDRColorCode_t *pOutput)
- int pack_nas_SLQSNasGetTxRxInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasGetTxRxInfo_t *reqArg)
- int unpack_nas_SLQSNasGetTxRxInfo (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGetTxRxInfo_t *pOutput)
- int pack_nas_SLQSGetOperatorNameData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetOperatorNameData (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetOperatorNameData_t *pOutput)
- int pack_nas_SLQSNasGet3GPP2Subscription (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNasGet3GPP2Subscription_t *reqArg)
- int unpack_nas_SLQSNasGet3GPP2Subscription (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNasGet3GPP2Subscription_t *pOutput)

- int pack_nas_SLQSSwiGetHDRPersonality (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetHDRPersonality (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetHDRPersonality_t *pOutput)
- int pack_nas_SLQSSwiGetHDRProtSubtype (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetHDRProtSubtype (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetHDRProtSubtype_t *pOutput)
- int pack_nas_SLQSSwiPSDetach (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSSwiPSDetach_t *pReqParam)
- int unpack_nas_SLQSSwiPSDetach (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSGetErrorRate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSGetErrorRate (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSGetErrorRate_t *pOutput)
- int pack_nas_SLQSSwiGetHRPDStats (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiGetHRPDStats (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiGetHRPDStats_t *pOutput)
- int pack_nas_SLQSSwiNetworkDebug (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSSwiNetworkDebug (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSSwiNetworkDebug_t *pOutput)
- int pack_nas_SLQSConfigSigInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSConfigSigInfo_t *pReqParam)
- int unpack_nas_SLQSConfigSigInfo (uint8_t *pResp, uint16_t respLen)
- int pack_nas_GetHomeNetwork3GPP2 (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_GetHomeNetwork3GPP2 (uint8_t *pResp, uint16_t respLen, unpack_nas_GetHomeNetwork3GPP2_t *pOutput)
- int pack_nas_SLQSNASSwiSetChannelLock (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_nas_SLQSNASSwiSetChannelLock_t *pReqParam)
- int unpack_nas_SLQSNASSwiSetChannelLock (uint8_t *pResp, uint16_t respLen)
- int pack_nas_SLQSNASSwiGetChannelLock (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_nas_SLQSNASSwiGetChannelLock (uint8_t *pResp, uint16_t respLen, unpack_nas_SLQSNASSwiGetChannelLock_t *pOutput)

9.8.1 Macro Definition Documentation

9.8.1.1 #define NAS_IMSI_M_S1_LENGTH 7

9.8.1.2 #define NAS_IMSI_M_S2_LENGTH 3

9.8.1.3 #define NAS_MAX_DESCRIPTION_LENGTH 255

9.8.1.4 #define NAS_MAX_NUM_NETWORKS 30

9.8.1.5 #define NAS_MAX_SCC_RX_INFO_INSTANCES 255

9.8.1.6 #define NAS_NAM_NAME_LENGTH 12

9.8.1.7 #define NAS_OTA_MESSAGE_MAX_BUF_SIZE 2048

9.8.1.8 #define NAS_PLMN_LENGTH 3

9.8.1.9 #define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST 255

9.8.2 Enumeration Type Documentation

9.8.2.1 enum LITEQMI_NAS_LTE_CPHY_CA_BW_NRB

Enumerator

eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6

eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15

eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25

eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50

eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75

eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100

9.8.2.2 enum LITEQMI_NAS_LTE_CPHY_SCELL_STATE

Enumerator

eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED

eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED

eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED

9.8.2.3 enum NAS_LTE_CPHY_CA_BW_NRB_LITE

Enumerator

eNAS_LTE_CPHY_CA_BW_NRB_LITE_6

eNAS_LTE_CPHY_CA_BW_NRB_LITE_15

eNAS_LTE_CPHY_CA_BW_NRB_LITE_25

eNAS_LTE_CPHY_CA_BW_NRB_LITE_50

eNAS_LTE_CPHY_CA_BW_NRB_LITE_75

eNAS_LTE_CPHY_CA_BW_NRB_LITE_100

9.8.2.4 enum NAS_LTE_CPHY_SCELL_STATE_LITE

Enumerator

eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE

eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE

eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE

9.8.3 Function Documentation

9.8.3.1 int pack_nas_GetACCOLC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Retrieves information about the access overload class (ACCOLC) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.2 int pack_nas_GetANAAAAuthenticationStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

AN-AAA authentication status of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.3 int pack_nas_GetCDMANetworkParameters (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.4 int pack_nas_GetHomeNetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get home network pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.5 int pack_nas_GetHomeNetwork3GPP2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Home Network 3GPP2 Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.6 int pack_nas_GetNetworkPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

9.8.3.7 int pack_nas_GetRFInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get rf info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.8 int pack_nas_GetServingNetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.9 int pack_nas_GetServingNetworkCapabilities (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.10 int pack_nas_GetSignalStrengths (pack_qmi_t * *pCtx*, uint8_t * *pReq*, uint16_t * *pLen*)

get signal strengths pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.11 int pack_nas_InitiateDomainAttach (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_InitiateDomainAttach_t * *pReqParam*)

Initiates a domain attach/detach of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.12 int pack_nas_PerformNetworkScan (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.13 int pack_nas_SetACCOLC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SetACCOLC_t *reqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.14 int pack_nas_SetCDMANetworkParameters (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SetCDMANetworkParameters_t * *pReqParam*)

CDMA Network Parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.15 int pack_nas_SetLURejectCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint8_t * *pBenable*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pBenable*	0/1 value to disable/enable indication respectively

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.16 int pack_nas_SetNetworkPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SetNetworkPreference_t * *reqArg*)

Sets the network registration preference. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use pack_nas_SLQSSetSysSelectionPref() for new firmware versions and new modules

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

9.8.3.17 int pack_nas_SetRFInfoCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint8_t * *pBenable*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pBenable*	0/1 value to disable/enable indication respectively

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.18 int pack_nas_SLQSConfigSigInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSConfigSigInfo_t * *pReqParam*)

Config Sig Info Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.19 int pack_nas_SLQSGetErrorRate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Error Rate Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.20 int pack_nas_SlqsGetLTECphyCAInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

9.8.3.21 int pack_nas_SLQSGetNetworkTime (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.22 int pack_nas_SLQSGetOperatorNameData (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Operator Name Data Parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.23 int pack_nas_SLQSGetPLMNName (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_nas_SLQSGetPLMNName_t * reqArg)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.24 int pack_nas_SLQSGetServingSystem (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.25 int pack_nas_SLQSGetSignalStrength (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint16_t *reqMask*)

Queries the current signal strength as measured by the device. This API is deprecated on MC73xx/EM73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use unpack_nas_SLQSNasGetSigInfo() for new firmware versions and new modules

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqMask*	request mask for fetching extra signal info

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.26 int pack_nas_SLQSGetSysInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.27 int pack_nas_SLQSGetSysSelectionPref (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Queries the different system selection preferences of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.28 int pack_nas_SLQSInitiateNetworkRegistration (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSInitiateNetworkRegistration_t * *pReqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.29 int pack_nas_SLQSNasConfigSigInfo2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasConfigSigInfo2_t * *pReqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.30 int pack_nas_SLQSNasGet3GPP2Subscription (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasGet3GPP2Subscription_t * *reqArg*)

3GPP2 Subscription Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.31 int pack_nas_SLQSNasGetCellLocationInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.32 int pack_nas_SLQSNasGetHDRColorCode (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HDR Color Code Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.33 int pack_nas_SLQSNasGetSigInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get sig info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.34 int pack_nas_SLQSNasGetTxRxInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasGetTxRxInfo_t * *reqArg*)

Get Tx Rx Info Parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.35 int pack_nas_SLQSNasIndicationRegisterExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasIndicationRegisterExt_t * *pReqParam*)

Used to Registers/De-registers for different NAS (Network access service) indications.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This function is used by a control point to register/deregister for different QMI_NAS indications. The control point's registration state variables, controlling registration for indications, are modified to reflect the settings indicated in the parameters that are present in the request message. At least one optional parameter must be present in the request.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.36 int pack_nas_SLQSNASSwiGetChannelLock (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get Channel Lock Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.37 int pack_nas_SLQSNasSwiIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasSwiIndicationRegister_t * *pReqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.38 int pack_nas_SLQSNasSwiModemStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

This function requests the device to return the current status of modem.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.39 int pack_nas_SLQSNASSwiSetChannelLock (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNASSwiSetChannelLock_t * *pReqParam*)

Set Channel Lock Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.40 int pack_nas_SLQSSetBandPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSetBandPreference_t * *pReqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.41 int pack_nas_SLQSSetSignalStrengthsCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSetSignalStrengthsCallback_t * *pReqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request prarmeters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.42 int pack_nas_SLQSSetSysSelectionPref (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSetSysSelectionPref_t * *pReqParam*)

Sets the different system selection preferences of the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values4780

9.8.3.43 int pack_nas_SLQSSwiGetHDRPersonality (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HDR Personality Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.44 int pack_nas_SLQSSwiGetHDRProtSubtype (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HDR Protocol Sub type Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.45 int pack_nas_SLQSSwiGetHRPDStats (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

HRPD Stats Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.46 int pack_nas_SLQSSwiGetLteCQI (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.47 int pack_nas_SLQSSwiGetLteSccRxInfo (pack_qmi_t * *pCtx*, uint8_t * *pReq*, uint16_t * *pLen*)

get LTE Secondary carrier Rx signal level information pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.48 int pack_nas_SLQSSwiNetworkDebug (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Network Debug Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.49 int pack_nas_SLQSSwiPSDetach (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSwiPSDetach_t * *pReqParam*)

PS Connection Detach Parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request Parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.50 int unpack_nas_GetACCOLC (uint8_t * *pResp*, uint16_t *respLen*, uint8_t * *pAccolc*)

Retrieves information about the access overload class (ACCOLC) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pAccolc*	accolc

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.51 int unpack_nas_GetANAAAAuthenticationStatus (uint8_t * *pResp*, uint16_t *respLen*, uint32_t * *pAuthStatus*)

AN-AAA authentication status of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pAuthStatus*	auth status

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.52 int unpack_nas_GetCDMANetworkParameters (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetCDMANetworkParameters_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	qmi output parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.53 int unpack_nas_GetHomeNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetHomeNetwork_t * *pOutput*)

Retrieves information about the home network of the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.54 int unpack_nas_GetHomeNetwork3GPP2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetHomeNetwork3GPP2_t * *pOutput*)

Get Home Network 3GPP2 Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.55 int unpack_nas_GetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetNetworkPreference_t * *pOutput*)

Sets the network registration preference unpack

Parameters

in	*pResp*	qmi resp from modem
in	*respLen*	qmi resp length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.56 int unpack_nas_GetRFInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetRFInfo_t * *pOutput*)

get rf info unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.57 int unpack_nas_GetServingNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetwork_t * *pOutput*)

Provides information about the system that provides service to the device. This API is deprecated on MC73xx/E-M73xx modules since firmware version SWI9X15C_05_xx_xx_xx and all EM74xx firmware versions. Please use API unpack_nas_SLQSGetSysInfo() for new firmware versions and new modules

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.58 int unpack_nas_GetServingNetworkCapabilities (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetworkCapabilities_t * *pOutput*)

Returns information regarding the data capabilities of the system that currently provides service to the device.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.59 int unpack_nas_GetSignalStrengths (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetSignalStrengths_t * *pOutput*)

Returns the available signal strengths (in dBm) as measured by the device in an array. This will also provides the corresponding radio interfaces in an array.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.60 int unpack_nas_InitiateDomainAttach (uint8_t * *pResp*, uint16_t *respLen*)

Initiates a domain attach/detach of the device unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.61 int unpack_nas_PerformNetworkScan (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_PerformNetworkScan_t * *pOutput*)

Performs scan for available networks and scans for RAT info as well.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.62 int unpack_nas_SetACCOLC (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.63 int unpack_nas_SetCDMANetworkParameters (uint8_t * *pResp*, uint16_t *respLen*)

CDMA Network Parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.64 int unpack_nas_SetDataCapabilitiesCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetDataCapabilitiesCallback_ind_t * *pOutput*)

Data Capabilities indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.65 int unpack_nas_SetEventReportInd (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetEventReportInd_t * *pOutput*)

9.8.3.66 int unpack_nas_SetLURejectCallback (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.67 int unpack_nas_SetNasLTECphyCaIndCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetNasLTECphyCaIndCallback_ind_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.68 int unpack_nas_SetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetNetwork-Preference_t * *pOutput*)

Sets the network registration preference unpack

Parameters

in	*pResp*	qmi resp from modem
in	*respLen*	qmi resp length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.69 int unpack_nas_SetRFInfoCallback (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.70 int unpack_nas_SetRoamingIndicatorCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetRoamingIndicatorCallback_ind_t * *pOutput*)

Roaming indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.71 int unpack_nas_SetServingSystemCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetServingSystemCallback_ind_t * *pOutput*)

Set Serving system indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.72 int unpack_nas_SLQSConfigSigInfo (uint8_t * *pResp*, uint16_t *respLen*)

Config Sig Info Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.73 int unpack_nas_SLQSGetErrorRate (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetErrorRate_t * *pOutput*)

Get Error Rate Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.74 int unpack_nas_SlqsGetLTECphyCAInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SlqsGetLTECphyCAInfo_t * *pOutput*)

Get LTE CPHY Carrier Info unpack

Parameters

in	*pResp*	qmi resp from modem
in	*respLen*	qmi resp length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.75 int unpack_nas_SLQSGetNetworkTime (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetNetworkTime_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.76 int unpack_nas_SLQSGetOperatorNameData (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetOperatorNameData_t * *pOutput*)

Operator Name Data Parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.77 int unpack_nas_SLQSGetPLMNName (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetPLMNName_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.78 int unpack_nas_SLQSGetServingSystem (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetServingSystem_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.79 int unpack_nas_SLQSGetSignalStrength (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSignalStrength_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.80 int unpack_nas_SLQSGetSysInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysInfo_t * *pOutput*)

Provides the system information. This API is preferred when trying to get the service status info and serving system info. The function unpack_nas_SLQSGetServingSystem_t() reports similar NAS information, but it is deprecated. Please refer to the header description of unpack_nas_SLQSGetServingSystem_t() for more information.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This API queries current serving system information, including registration information and system property. The registration information for all RATs specified in the mode capability setting are included regardless of registration status. The RAT-specific system property are included only for RATs that are specified in the mode capability setting and which are not in either No Service or Power Save modes.

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.81 int unpack_nas_SLQSGetSysSelectionPref (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysSelectionPref_t * *pOutput*)

Queries the different system selection preferences of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.82 int unpack_nas_SLQSInitiateNetworkRegistration (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.83 int unpack_nas_SLQSNasConfigSigInfo2 (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.84 int unpack_nas_SLQSNasGet3GPP2Subscription (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGet3GPP2Subscription_t * *pOutput*)

3GPP2 Subscription Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.85 int unpack_nas_SLQSNasGetCellLocationInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetCellLocationInfo_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.86 int unpack_nas_SLQSNasGetHDRColorCode (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetHDRColorCode_t * *pOutput*)

HDR Color Code Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.87 int unpack_nas_SLQSNasGetSigInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetSigInfo_t * *pOutput*)

get sig info unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

Note

This command queries the signal strength information for currently active RATs. Information is reported only if the corresponding RATs have signal strength values to be reported. If no signal strength information is available for any RAT, the response message contains only the mandatory response message

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.88 int unpack_nas_SLQSNasGetTxRxInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasGetTxRxInfo_t * *pOutput*)

Get Tx Rx Info Parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.89 int unpack_nas_SLQSNasIndicationRegisterExt (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.90 int unpack_nas_SLQSNasNetworkTimeCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasNetworkTimeCallBack_ind_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.91 int unpack_nas_SLQSNasSigInfoCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSigInfoCallback_ind_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	sig info indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.92 int unpack_nas_SLQSNASSwiGetChannelLock (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNASSwiGetChannelLock_t * *pOutput*)

Get Channel Lock Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.93 int unpack_nas_SLQSNasSwiIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.94 int unpack_nas_SLQSNasSwiModemStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSwiModemStatus_t * *pOutput*)

This function requests the device to return the current status of modem.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.95 int unpack_nas_SLQSNasSwiOTAMessageCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t * *pOutput*)

OTA message indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.96 int unpack_nas_SLQSNASSwiSetChannelLock (uint8_t * *pResp*, uint16_t *respLen*)

Set Channel Lock Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.97 int unpack_nas_SLQSNasSysInfoCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSysInfoCallback_ind_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.98 int unpack_nas_SLQSNasTimerCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasTimerCallback_ind_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	network timer indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.99 int unpack_nas_SLQSSetBandPreference (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

9.8.3.100 int unpack_nas_SLQSSetSignalStrengthsCallback (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.101 int unpack_nas_SLQSSetSysSelectionPref (uint8_t * *pResp*, uint16_t *respLen*)

Sets the different system selection preferences of the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.102 int unpack_nas_SLQSSetSysSelectionPrefCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t * *pOutput*)

System Selection Preference indication unpack

Parameters

in	*pResp*	qmi indication from modem
in	*respLen*	qmi indication length
out	*pOutput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.103 int unpack_nas_SLQSSwiGetHDRPersonality (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetHDRPersonality_t * *pOutput*)

HDR Personality Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.104 int unpack_nas_SLQSSwiGetHDRProtSubtype (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetHDRProtSubtype_t * *pOutput*)

HDR Protocol Sub type Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.105 int unpack_nas_SLQSSwiGetHRPDStats (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetHRPDStats_t * *pOutput*)

HRPD Stats Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.106 int unpack_nas_SLQSSwiGetLteCQI (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetLteCQI_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.107 int unpack_nas_SLQSSwiGetLteSccRxInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiGetLteSccRxInfo_t * *pOutput*)

get LTE Secondary carrier Rx signal level information unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.108 int unpack_nas_SLQSSwiNetworkDebug (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSSwiNetworkDebug_t * *pOutput*)

Network Debug Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.3.109 int unpack_nas_SLQSSwiPSDetach (uint8_t * *pResp*, uint16_t *respLen*)

PS Connection Detach Parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9 pds.h File Reference

Data Structures

- struct unpack_pds_GetPDSState_t
- struct pack_pds_SetPDSState_t
- struct unpack_pds_GetPDSDefaults_t
- struct pack_pds_SetPDSDefaults_t
- struct unpack_pds_GetPortAutomaticTracking_t
- struct pack_pds_SetPortAutomaticTracking_t
- struct pack_pds_StartPDSTrackingSessionExt_t
- struct pack_pds_PDSInjectTimeReference_t
- struct unpack_pds_GetXTRAAutomaticDownload_t
- struct pack_pds_SetXTRAAutomaticDownload_t
- struct unpack_pds_GetXTRANetwork_t
- struct pack_pds_SetXTRANetwork_t
- struct unpack_pds_GetXTRAValidity_t
- struct unpack_pds_GetServiceAutomaticTracking_t
- struct pack_pds_SetServiceAutomaticTracking_t
- struct pack_pds_ResetPDSData_t
- struct pack_pds_SLQSSetAGPSConfig_t
- struct pack_pds_SLQSPDSInjectAbsoluteTimeReference_t
- struct pack_pds_SLQSGetAGPSConfig_t
- struct unpack_pds_SLQSGetAGPSConfig_t
- struct pack_pds_SLQSPDSInjectPositionData_t
- struct unpack_pds_SLQSGetGPSStateInfo_t
- struct pack_pds_SLQSSetPositionMethodState_t
- struct pack_pds_SetEventReportCallback_t
- struct unpack_pds_SetEventReport_Ind_t
- struct unpack_pds_SetPdsState_Ind_t

Functions

- int pack_pds_GetPDSState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetPDSState (uint8_t *pResp, uint16_t respLen, unpack_pds_GetPDSState_t *pOutput)
- int pack_pds_SetPDSState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetPDSState_t *pReqParam)
- int unpack_pds_SetPDSState (uint8_t *pResp, uint16_t respLen)
- int pack_pds_GetPDSDefaults (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetPDSDefaults (uint8_t *pResp, uint16_t respLen, unpack_pds_GetPDSDefaults_t *pOutput)

- int pack_pds_SetPDSDefaults (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetPDSDefaults_t *pReqParam)
- int unpack_pds_SetPDSDefaults (uint8_t *pResp, uint16_t respLen)
- int pack_pds_GetPortAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetPortAutomaticTracking (uint8_t *pResp, uint16_t respLen, unpack_pds_GetPortAutomaticTracking_t *pOutput)
- int pack_pds_SetPortAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetPortAutomaticTracking_t *pReqParam)
- int unpack_pds_SetPortAutomaticTracking (uint8_t *pResp, uint16_t respLen)
- int pack_pds_StartPDSTrackingSessionExt (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_StartPDSTrackingSessionExt_t *pReqParam)
- int unpack_pds_StartPDSTrackingSessionExt (uint8_t *pResp, uint16_t respLen)
- int pack_pds_StopPDSTrackingSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_StopPDSTrackingSession (uint8_t *pResp, uint16_t respLen)
- int pack_pds_PDSInjectTimeReference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_PDSInjectTimeReference_t *pReqParam)
- int unpack_pds_PDSInjectTimeReference (uint8_t *pResp, uint16_t respLen)
- int pack_pds_GetXTRAAutomaticDownload (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetXTRAAutomaticDownload (uint8_t *pResp, uint16_t respLen, unpack_pds_GetXTRAAutomaticDownload_t *pOutput)
- int pack_pds_SetXTRAAutomaticDownload (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetXTRAAutomaticDownload_t *pReqParam)
- int unpack_pds_SetXTRAAutomaticDownload (uint8_t *pResp, uint16_t respLen)
- int pack_pds_GetXTRANetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetXTRANetwork (uint8_t *pResp, uint16_t respLen, unpack_pds_GetXTRANetwork_t *pOutput)
- int pack_pds_SetXTRANetwork (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetXTRANetwork_t *pReqParam)
- int unpack_pds_SetXTRANetwork (uint8_t *pResp, uint16_t respLen)
- int pack_pds_GetXTRAValidity (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetXTRAValidity (uint8_t *pResp, uint16_t respLen, unpack_pds_GetXTRAValidity_t *pOutput)
- int pack_pds_ForceXTRADownload (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_ForceXTRADownload (uint8_t *pResp, uint16_t respLen)
- int pack_pds_GetServiceAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_GetServiceAutomaticTracking (uint8_t *pResp, uint16_t respLen, unpack_pds_GetServiceAutomaticTracking_t *pOutput)
- int pack_pds_SetServiceAutomaticTracking (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SetServiceAutomaticTracking_t *pReqParam)
- int unpack_pds_SetServiceAutomaticTracking (uint8_t *pResp, uint16_t respLen)
- int pack_pds_ResetPDSData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_ResetPDSData_t *pReqParam)
- int unpack_pds_ResetPDSData (uint8_t *pResp, uint16_t respLen)
- int pack_pds_SLQSSetAGPSConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSSetAGPSConfig_t *pReqParam)
- int unpack_pds_SLQSSetAGPSConfig (uint8_t *pResp, uint16_t respLen)
- int pack_pds_SLQSPDSInjectAbsoluteTimeReference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSPDSInjectAbsoluteTimeReference_t *pReqParam)
- int unpack_pds_SLQSPDSInjectAbsoluteTimeReference (uint8_t *pResp, uint16_t respLen)
- int pack_pds_SLQSGetAGPSConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSGetAGPSConfig_t *pReqParam)
- int unpack_pds_SLQSGetAGPSConfig (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSGetAGPSConfig_t *pOutput)
- int pack_pds_SLQSPDSInjectPositionData (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSPDSInjectPositionData_t *pReqParam)
- int unpack_pds_SLQSPDSInjectPositionData (uint8_t *pResp, uint16_t respLen)

- int pack_pds_SLQSPDSDeterminePosition (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_SLQSPDSDeterminePosition (uint8_t *pResp, uint16_t respLen)
- int pack_pds_SLQSGetGPSStateInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_pds_SLQSGetGPSStateInfo (uint8_t *pResp, uint16_t respLen, unpack_pds_SLQSGetGPSStateInfo_t *pOutput)
- int pack_pds_SLQSSetPositionMethodState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_pds_SLQSSetPositionMethodState_t *pReqParam)
- int unpack_pds_SLQSSetPositionMethodState (uint8_t *pResp, uint16_t respLen)
- int pack_pds_SetEventReportCallback (pack_qmi_t *pCtx, uint8_t *pReq, uint16_t *pLen, pack_pds_SetEventReportCallback_t *pReqParam)
- int unpack_pds_SetEventReportCallback (uint8_t *pResp, uint16_t respLen)
- int unpack_pds_SetEventReport_Ind (uint8_t *pResp, uint16_t respLen, unpack_pds_SetEventReport_Ind_t *pOutput)
- int unpack_pds_SetPdsState_Ind (uint8_t *pResp, uint16_t respLen, unpack_pds_SetPdsState_Ind_t *pOutput)

9.9.1 Function Documentation

9.9.1.1 int pack_pds_ForceXTRADownload (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Forces the XTRA database to be downloaded to the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.2 int pack_pds_GetPDSDefaults (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get the default tracking session configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.3 int pack_pds_GetPDSState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Get current PDS state pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.4 int pack_pds_GetPortAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Returns the automatic tracking configuration for the NMEA COM port pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.5 int pack_pds_GetServiceAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

automatic tracking state for the service pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.6 int pack_pds_GetXTRAAutomaticDownload (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

XTRA automatic database download configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.7 int pack_pds_GetXTRANetwork (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

XTRA WWAN network preference parameter pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.8 int pack_pds_GetXTRAValidity (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

XTRA database validity period parameter pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.9 int pack_pds_PDSInjectTimeReference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_PDSInjectTimeReference_t * *pReqParam*)

Inject system time parameters pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.10 int pack_pds_ResetPDSData (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_ResetPDSData_t * *pReqParam*)

Resets the specified PDS data pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.11 int pack_pds_SetEventReportCallback (pack_qmi_t * *pCtx*, uint8_t * *pReq*, uint16_t * *pLen*, pack_pds_SetEventReportCallback_t * *pReqParam*)

Enables/disables the PDS Event report callback function pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.12 int pack_pds_SetPDSDefaults (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetPDSDefaults_t * *pReqParam*)

Sets the default tracking session configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.13 int pack_pds_SetPDSState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetPDSState_t * *pReqParam*)

Sets the PDS state pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.14 int pack_pds_SetPortAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetPortAutomaticTracking_t * *pReqParam*)

Sets the automatic tracking configuration for the NMEA COM port pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.15 int pack_pds_SetServiceAutomaticTracking (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetServiceAutomaticTracking_t * *pReqParam*)

Sets the automatic tracking state for the service pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.16 int pack_pds_SetXTRAAutomaticDownload (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetXTRAAutomaticDownload_t * *pReqParam*)

Sets the XTRA automatic database download configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.17 int pack_pds_SetXTRANetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SetXTRANetwork_t * *pReqParam*)

Sets the XTRA WWAN network preference pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.18 int pack_pds_SLQSGetAGPSConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSGetAGPSConfig_t * *pReqParam*)

Gets the PDS AGPS (MS-based) configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.19 int pack_pds_SLQSGetGPSStateInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Queries the MSM GPS server for receiver state information pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.20 int pack_pds_SLQSPDSDeterminePosition (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Requests the MSM GPS service to obtain the current position for manually controlled tracking sessions pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.21 int pack_pds_SLQSPDSInjectAbsoluteTimeReference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSPDSInjectAbsoluteTimeReference_t * *pReqParam*)

Injects a absolute time reference into the PDS engine pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.22 int pack_pds_SLQSPDSInjectPositionData (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSPDSInjectPositionData_t * *pReqParam*)

Injects position data into the PDS engine pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.23 int pack_pds_SLQSSetAGPSConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSSetAGPSConfig_t * *pReqParam*)

Sets the PDS AGPS (MS-based) configuration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.24 int pack_pds_SLQSSetPositionMethodState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_SLQSSetPositionMethodState_t * *pReqParam*)

Sets the state of positioning methods for the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.25 int pack_pds_StartPDSTrackingSessionExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_pds_StartPDSTrackingSessionExt_t * *pReqParam*)

Starts a PDS tracking session pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameters

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.26 int pack_pds_StopPDSTrackingSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Stops a PDS tracking session pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.27 int unpack_pds_ForceXTRADownload (uint8_t * *pResp*, uint16_t *respLen*)

Forces the XTRA database to be downloaded to the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.28 int unpack_pds_GetPDSDefaults (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetPDSDefaults_t * *pOutput*)

Get the default tracking session configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.29 int unpack_pds_GetPDSState (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetPDSState_t * *pOutput*)

Get current PDS state unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.30 int unpack_pds_GetPortAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetPortAutomaticTracking_t * *pOutput*)

Returns the automatic tracking configuration for the NMEA COM port unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.31 int unpack_pds_GetServiceAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetServiceAutomaticTracking_t * *pOutput*)

automatic tracking state for the service unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.32 int unpack_pds_GetXTRAAutomaticDownload (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetXTRAAutomaticDownload_t * *pOutput*)

XTRA automatic database download configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.33 int unpack_pds_GetXTRANetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetXTRANetwork_t * *pOutput*)

XTRA WWAN network preference parameter unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.34 int unpack_pds_GetXTRAValidity (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_GetXTRAValidity_t * *pOutput*)

XTRA database validity period parameter unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.35 int unpack_pds_PDSInjectTimeReference (uint8_t * *pResp*, uint16_t *respLen*)

Inject system time parameters unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.36 int unpack_pds_ResetPDSData (uint8_t * *pResp*, uint16_t *respLen*)

Resets the specified PDS data unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.37 int unpack_pds_SetEventReport_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetEventReport_Ind_t * *pOutput*)

PDS Event Report Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.38 int unpack_pds_SetEventReportCallback (uint8_t * *pResp*, uint16_t *respLen*)

Enables/disables the PDS Event report callback function unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.39 int unpack_pds_SetPDSDefaults (uint8_t * *pResp*, uint16_t *respLen*)

Sets the default tracking session configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.40 int unpack_pds_SetPDSState (uint8_t * *pResp*, uint16_t *respLen*)

Sets the PDS state unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.41 int unpack_pds_SetPdsState_Ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SetPdsState_Ind_t * *pOutput*)

PDS State Indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.42 int unpack_pds_SetPortAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*)

Sets the automatic tracking configuration for the NMEA COM port unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.43 int unpack_pds_SetServiceAutomaticTracking (uint8_t * *pResp*, uint16_t *respLen*)

Sets the automatic tracking state for the service unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.44 int unpack_pds_SetXTRAAutomaticDownload (uint8_t * *pResp*, uint16_t *respLen*)

Sets the XTRA automatic database download configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.45 int unpack_pds_SetXTRANetwork (uint8_t * *pResp*, uint16_t *respLen*)

Sets the XTRA WWAN network preference unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.46 int unpack_pds_SLQSGetAGPSConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSGetAGPS-Config_t * *pOutput*)

Gets the PDS AGPS (MS-based) configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.47 int unpack_pds_SLQSGetGPSStateInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_pds_SLQSGetGPSStateInfo_t * *pOutput*)

Queries the MSM GPS server for receiver state information unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.48 int unpack_pds_SLQSPDSDeterminePosition (uint8_t * *pResp*, uint16_t *respLen*)

Requests the MSM GPS service to obtain the current position for manually controlled tracking sessions unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.49 int unpack_pds_SLQSPDSInjectAbsoluteTimeReference (uint8_t * *pResp*, uint16_t *respLen*)

Injects a absolute time reference into the PDS engine unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.50 int unpack_pds_SLQSPDSInjectPositionData (uint8_t * *pResp*, uint16_t *respLen*)

Injects position data into the PDS engine unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.51 int unpack_pds_SLQSSetAGPSConfig (uint8_t * *pResp*, uint16_t *respLen*)

Sets the PDS AGPS (MS-based) configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.52 int unpack_pds_SLQSSetPositionMethodState (uint8_t * *pResp*, uint16_t *respLen*)

Sets the state of positioning methods for the device unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.53 int unpack_pds_StartPDSTrackingSessionExt (uint8_t * *pResp*, uint16_t *respLen*)

Starts a PDS tracking session unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.9.1.54 int unpack_pds_StopPDSTrackingSession (uint8_t * *pResp*, uint16_t *respLen*)

Stops a PDS tracking session unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.10 qaGobiApiTableBandClasses.h File Reference

Network Access Service API Band Classes table.

9.10.1 Detailed Description

Network Access Service API Band Classes table.

9.10.2 Band Classes (Value - Description)

- 0 - CDMA Band Class 0

- 1 - CDMA Band Class 1
- 3 - CDMA Band Class 3
- 4 - CDMA Band Class 4
- 5 - CDMA Band Class 5
- 6 - CDMA Band Class 6
- 7 - CDMA Band Class 7
- 8 - CDMA Band Class 8
- 9 - CDMA Band Class 9
- 10 - CDMA Band Class 10
- 11 - CDMA Band Class 11
- 12 - CDMA Band Class 12
- 13 - CDMA Band Class 13
- 14 - CDMA Band Class 14
- 15 - CDMA Band Class 15
- 16 - CDMA Band Class 16
- 17 - CDMA Band Class 17
- 18 - CDMA Band Class 18
- 19 - CDMA Band Class 19
- 40 - GSM 450
- 41 - GSM 480
- 42 - GSM 750
- 43 - GSM 850
- 44 - GSM 900 (Extended)
- 45 - GSM 900 (Primary)
- 46 - GSM 900 (Railways)
- 47 - GSM 1800
- 48 - GSM 1900
- 80 - WCDMA 2100
- 81 - WCDMA PCS 1900
- 82 - WCDMA DCS 1800
- 83 - WCDMA 1700 (US)
- 84 - WCDMA 850
- 85 - WCDMA 800
- 86 - WCDMA 2600
- 87 - WCDMA 900
- 88 - WCDMA 1700 (Japan)

- 90 - WCDMA 1500 band (Japan)
- 91 - WCDMA 850 band (Japan)
- < Reserved 89, 92-109 for WCDMA band classes>
- 110 - WLAN US 2400 MHz
- 111 - WLAN JAPAN 2400 MHz
- 112 - WLAN EUROPEAN 2400 MHz
- 113 - WLAN FRANCE 2400 MHz
- 114 - WLAN SPAIN 2400 MHz
- 115 - WLAN US 5000 MHz band
- 116 - WLAN JAPAN 5000 MHz
- 117 - WLAN EUROPEAN 5000 MHz
- 118 - WLAN FRANCE 5000 MHz
- 119 - WLAN SPAIN 5000 MHz

9.10.2.1 LTE Bands

- 28 - LTE Band Class 28
- 39 - LTE Band Class 39
- 40 - LTE Band Class 40
- 41 - LTE Band Class 41
- 120 - FDD UL:1920-1980; DL:2110-2170; E-UTRA Operating Band 1
- 121 - FDD UL:1850-1910; DL:1930-1990; E-UTRA Operating Band 2
- 122 - FDD UL:1710-1785; DL:1805-1880; E-UTRA Operating Band 3
- 123 - FDD UL:1710-1755; DL:2110-2155; E-UTRA Operating Band 4
- 124 - FDD UL: 824- 849; DL: 869- 894; E-UTRA Operating Band 5
- 125 - FDD UL: 830- 840; DL: 875- 885; E-UTRA Operating Band 6
- 126 - FDD UL:2500-2570; DL:2620-2690; E-UTRA Operating Band 7
- 127 - FDD UL: 880- 915; DL: 925- 960; E-UTRA Operating Band 8
- 128 - FDD UL:1749.9-1784.9; DL:1844.9-1879.9; E-UTRA Operating Band 9
- 129 - FDD UL:1710-1770; DL:2110-2170; E-UTRA Operating Band 10
- 130 - FDD UL:1427.9-1452.9; DL:1475.9-1500.9; E-UTRA Operating Band 11
- 131 - FDD UL:698-716; DL:728-746; E-UTRA Operating Band 12
- 132 - FDD UL: 777- 787; DL: 746-756; E-UTRA Operating Band 13
- 133 - FDD UL: 788- 798; DL: 758-768; E-UTRA Operating Band 14
- 134 - FDD UL: 704-716; DL: 734-746; E-UTRA Operating Band 17
- 135 - TDD LTE UL: 1900-1920; DL: 1900-1920; E-UTRA Operating Band 33
- 136 - TDD LTE UL: 2010-2025; DL: 2010-2025; E-UTRA Operating Band 34

- 137 - TDD LTE UL: 1850-1910; DL: 1850-1910; E-UTRA Operating Band 35
- 138 - TDD LTE UL: 1930-1990; DL: 1930-1990; E-UTRA Operating Band 36
- 139 - TDD LTE UL: 1910-1930; DL: 1910-1930; E-UTRA Operating Band 37
- 140 - TDD LTE UL: 2570-2620; DL: 2570-2620; E-UTRA Operating Band 38
- 141 - TDD LTE UL: 1880-1920; DL: 1880-1920; E-UTRA Operating Band 39
- 142 - TDD LTE UL: 2300-2400; DL: 2300-2400; E-UTRA Operating Band 40
- 143 - FDD LTE UL: 815-830; DL: 860-875; E-UTRA Operating Band 18
- 144 - FDD LTE UL: 830-845; DL: 875-890; E-UTRA Operating Band 19
- 145 - FDD LTE UL: 832-862; DL: 791-821; E-UTRA Operating Band 20
- 146 - FDD LTE UL: 1447.9-1462.9; DL: 1495.9-1510.9; E-UTRA Operating Band 21
- 147 - FDD LTE UL: 1626.5-1660.5; DL: 1525-1559; E-UTRA Operating Band 24
- 148 - FDD LTE UL: 1850-19195; DL: 1930-1995; E-UTRA Operating Band 25
- 149 - TDD LTE UL: 2496-2690; DL: 2496-2690; E-UTRA Operating Band 41
- 150 - TDD LTE UL: 3400-3600; DL: 3400-3600; E-UTRA Operating Band 42
- 151 - TDD LTE UL: 3600-3800; DL: 3600-3800; E-UTRA Operating Band 43
- 200 - TD-SCDMA Band A
- 201 - TD-SCDMA Band B
- 202 - TD-SCDMA Band C
- 203 - TD-SCDMA Band D
- 204 - TD-SCDMA Band E
- 205 - TD-SCDMA Band F

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.11 qaGobiApiTableCallControlReturnReasons.h File Reference

Call Control Return Reasons table.

9.11.1 Detailed Description

Call Control Return Reasons table.

9.11.2 Call Control Result Reasons (Value - Name - Description)

- 0x01 - QMI_VOICE_REASON_FWD_UNCONDITIONAL - Unconditional call forwarding
- 0x02 - QMI_VOICE_REASON_FWD_MOBILEBUSY - Forward when the mobile is busy
- 0x03 - QMI_VOICE_REASON_FWD_NOREPLY - Forward when there is no reply
- 0x04 - QMI_VOICE_REASON_FWD_UNREACHABLE - Forward when the call is unreachable
- 0x05 - QMI_VOICE_REASON_FWD_ALLFORWARDING - All forwarding

- 0x06 - QMI_VOICE_REASON_FWD_ALLCONDITIONAL - All conditional forwarding
- 0x07 - QMI_VOICE_REASON_BARR_ALLOUTGOING - All outgoing
- 0x08 - QMI_VOICE_REASON_BARR_OUTGOINGINT - Outgoing internal
- 0x09 - QMI_VOICE_REASON_BARR_OUTGOINGINTEXTOHOME - Outgoing external to home
- 0x0A - QMI_VOICE_REASON_BARR_ALLINCOMING - All incoming
- 0x0B - QMI_VOICE_REASON_BARR_INCOMINGROAMING - Roaming incoming
- 0x0C - QMI_VOICE_REASON_BARR_ALLBARRING - All calls are barred
- 0x0D - QMI_VOICE_REASON_BARR_ALLOUTGOINGBARRING - All outgoing calls are barred
- 0x0E - QMI_VOICE_REASON_BARR_ALLINCOMINGBARRING - All incoming calls are barred
- 0x0F - QMI_VOICE_REASON_CALLWAITING - Call waiting
- 0x10 - VOICE_CC_SUPS_RESULT_REASON_CLIP - CLIP
- 0x11 - VOICE_CC_SUPS_RESULT_REASON_CLIR - CLIR
- 0x12 - VOICE_CC_SUPS_RESULT_REASON_COLP - COLP
- 0x13 - VOICE_CC_SUPS_RESULT_REASON_COLR - COLR
- 0x14 - VOICE_CC_SUPS_RESULT_REASON_CNAP - CNAP
- 0xFF - Not Available

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.12 qaGobiApiTableCallEndReasons.h File Reference

Wireless Data Service Call End Reasons.

9.12.1 Detailed Description

Wireless Data Service Call End Reasons.

9.12.2 Call end reason codes (Code - Reason)

9.12.2.1 Technology-agnostic call end reasons

- 1 - Reason unspecified, check the verbose call end reason
- 2 - Client ended the call
- 3 - Device has no service
- 4 - Call ended abnormally
- 5 - Received release from base station; no reason given
- 6 - Access attempt already in progress; SD2.0 only
- 7 - Access failure for reason other than the above
- 8 - Call rejected because of redirection or handoff
- 9 - Call failed because close is in progress

- 10 - Authentication failed, 3GPP equivalent ESM(EPS Session Management) cause code value 29, User authentication failed
- 11 - Call ended because of internal call end. This error code is returned when data call is brought down due to some unknown error, such as not specific to any RAT
- 12 - Call ended because of internal error. This error code is returned when data call is brought down due to some unspecified internal error, such as NULL pointer
- 13 - Internal unknown cause code

9.12.2.2 EVDO CDMA 1xEV-DO

- 500 - Device is CDMA-locked until power cycle
- 501 - Received intercept from base station; origination only
- 502 - Received reorder from base station; origination only
- 503 - Received release from base station; service option reject
- 504 - Received incoming call from base station
- 505 - Received alert stop from base station; incoming only
- 506 - Received end activation; OTASP call only
- 507 - Max access probes transmitted
- 508 - Concurrent service is not supported by base station
- 509 - No response received from base station
- 510 - Call rejected by the base station; CDMA only
- 511 - Concurrent services requested were not compatible; CDMA only
- 512 - Corresponds to CM CALL ORIG ERR ALREADY IN TC
- 513 - Used if Call manager subsystem is ending a GPS call in favor of a user call
- 514 - Used if Call manager subsystem is ending a SMS call in favor of a user call
- 515 - CDMA Only; Device has no service

9.12.2.3 WCDMA/GSM call end reasons

- 1000 - Call origination request failed; WCDMA/GSM Only
- 1001 - Client rejected the incoming call; WCDMA/GSM Only
- 1002 - Device has no UMTS service; WCDMA/GSM Only
- 1003 - Network ended the call, look in cc call; WCDMA/GSM Only
- 1004 - LLC(Logical Link Control) or SNDCP(Sub Network Dependent Convergence Protocol) failure
- 1005 - Insufficient resources, 3GPP equivalent ESM(EPS Session Management) cause code value 26, Insufficient resources
- 1006 - Service option temporarily out of order, 3GPP equivalent ESM(EPS Session Management) cause code value 34, Service option temporarily out of order
- 1007 - PTI already used, 3GPP equivalent ESM(EPS Session Management) cause code value 35, PTI(-Procedure Transaction Identity) already in use

- 1008 - Regular PDP context deactivation, 3GPP equivalent ESM(EPS Session Management) cause code value 36, Regular deactivation
- 1009 - Network failure, 3GPP equivalent ESM(EPS Session Management) cause code value 38, Network failure
- 1010 - Reactivation requested, 3GPP equivalent ESM(EPS Session Management) cause code value 39, Reactivation requested
- 1011 - Protocol error, unspecified, 3GPP equivalent ESM(EPS Session Management) cause code value 111, Protocol error, unspecified
- 1012 - Operator determined barring, 3GPP equivalent ESM(EPS Session Management) cause code value 8, Operator Determined Barring
- 1013 - Unknown or missing Access Point Name (APN), 3GPP equivalent ESM(EPS Session Management) cause code value 27, Missing or unknown APN
- 1014 - Unknown PDP address or PDP type, 3GPP equivalent ESM(EPS Session Management) cause code value 28, Unknown PDN type
- 1015 - Activation rejected by GGSN, 3GPP equivalent ESM(EPS Session Management) cause code value 30, Requested rejected by Serving GW or PDN GW
- 1016 - Activation rejected, unspecified, 3GPP equivalent ESM(EPS Session Management) cause code value 31, Request rejected, unspecified
- 1017 - Service option not supported, 3GPP equivalent ESM(EPS Session Management) cause code value 32, Service option not supported
- 1018 - Requested service option not subscribed, 3GPP equivalent ESM(EPS Session Management) cause code value 33, Requested service option not subscribed
- 1019 - EPS Quality of Service (QoS) not accepted, 3GPP equivalent ESM(EPS Session Management) cause code value 37, EPS QoS not accepted
- 1020 - Semantic error in the TFT operation, 3GPP equivalent ESM(EPS Session Management) cause code value 41, Semantic error in the TFT operation
- 1021 - Syntactical error in the TFT operation, 3GPP equivalent ESM(EPS Session Management) cause code value 42, Syntactical error in the TFT operation
- 1022 - Unknown PDP context, 3GPP equivalent ESM(EPS Session Management) cause code value 43, Invalid EPS bearer identity
- 1023 - Semantic errors in packet filter(s), 3GPP equivalent ESM(EPS Session Management) cause code value 44, Semantic errors in packet filter(s)
- 1024 - Syntactical error in packet filter(s), 3GPP equivalent ESM(EPS Session Management) cause code value 45, Syntactical errors in packet filter(s)
- 1025 - PDP context without TFT already activated, 3GPP equivalent ESM(EPS Session Management) cause code value 46, Unused
- 1026 - Invalid transaction identifier value, 3GPP equivalent ESM(EPS Session Management) cause code value 81, Invalid PTI value
- 1027 - Semantically incorrect message, 3GPP equivalent ESM(EPS Session Management) cause code value 95, Semantically incorrect message
- 1028 - Invalid mandatory information, 3GPP equivalent ESM(EPS Session Management) cause code value 96, Invalid mandatory information
- 1029 - Message type non-existent or not implemented, 3GPP equivalent ESM(EPS Session Management) cause code value 97, Message type non-existent or not implemented

- 1030 - Message not compatible with state, 3GPP equivalent ESM(EPS Session Management) cause code value 98, Message type not compatible with the protocol state
- 1031 - Information element nonexistent or not implemented, 3GPP equivalent ESM(EPS Session Management) cause code value 99, Information element non-existent or not implemented
- 1032 - Conditional information element error, 3GPP equivalent ESM(EPS Session Management) cause code value 100, Conditional IE error
- 1033 - Message not compatible with protocol state, 3GPP equivalent ESM(EPS Session Management) cause code value 101, Message not compatible with the protocol state
- 1034 - APN restriction value incompatible with active PDP context, 3GPP equivalent ESM(EPS Session Management) cause code value 112, APN restriction value incompatible with
  - active EPS bearer context
- 1035 - No GPRS context present
- 1036 - Requested feature not supported, 3GPP equivalent ESM(EPS Session Management) cause code value 40, Feature not supported
- 1037 - Illegal MS, 3GPP equivalent EMM(EPS Mobility Management) cause code value 3, Illegal UE (MS)
- 1038 - Illegal ME, 3GPP equivalent EMM(EPS Mobility Management) cause code value 6, Illegal ME. This error code is sent to the MS if the ME used is not acceptable
  - to the network, e.g. blacklisted
- 1039 - GPRS and non GPRS services not allowed
- 1040 - GPRS services not allowed
- 1041 - MS identity not derived by the network, 3GPP equivalent EMM(EPS Mobility Management) cause code value 9, UE (MS) Identify cannot be derived by the network
- 1042 - Implicitly detached, 3GPP equivalent EMM(EPS Mobility Management) cause code value 10, Implicitly Detached
- 1043 - PLMN not allowed, 3GPP equivalent EMM(EPS Mobility Management) cause code value 11, PLMN not allowed
- 1044 - LA not allowed, this cause is sent to the MS if it requests location updating in a location area where the HPLMN determines that the MS, by subscription, is not allowed to operate.
- 1045 - GPRS services not allowed in PLMN
- 1046 - PDP duplicate
- 1047 - UE radio access technology change
- 1048 - app preempted
- 1049 - Congestion, This cause is sent if the service request or LOCATION UPDATING REQUEST message cannot be actioned because of congestion (e.g. congestion of the MSC or SGSN or GGSN or PDN Gateway; no channel; facility busy/congested etc.).
- 1050 - No PDP context activated
- 1051 - Access class DSAC rejection

9.12.2.4 EVDO CDMA 1xEV-DO

- 1500 - Abort connection setup due to the reception of a Connection Deny message with deny code set to either general or network busy.
- 1501 - Abort connection setup due to the reception of a Connection Deny message with deny code set to either billing or authentication failure.
- 1502 - Change HDR system due to redirection or PRL not preferred
- 1503 - Exit HDR due to redirection or PRL not preferred
- 1504 - No HDR session
- 1505 - Used if Call manager is ending an HDR call origination in favor of a GPS fix
- 1506 - Connection setup timeout
- 1507 - Call manager released HDR call so 1x call can continue

9.12.2.5 call end reason type

- 1 - Mobile IP
- 2 - Internal
- 3 - Call Manager defined
- 6 - 3GPP specification defined
- 7 - PPP
- 8 - EHRPD
- 9 - IPv6

9.12.2.6 Mobile IP call end reasons (Type=1)

- 64 - MIP(Mobile IP) FA(Foreign Agent) ERR REASON UNSPECIFIED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration with unspecified reason
- 65 - MIP(Mobile IP) FA(Foreign Agent) ERR ADMINISTRATIVELY PROHIBITED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent administratively prohibited MIP registration
- 66 - MIP(Mobile IP) FA(Foreign Agent) ERR INSUFFICIENT RESOURCES, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to insufficient resources
- 67 - MIP(Mobile IP) FA(Foreign Agent) ERR MOBILE NODE AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because MN-AAA authenticator was wrong
- 68 - MIP(Mobile IP) FA(Foreign Agent) ERR HA AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because of home agent authentication failure
- 69 - MIP(Mobile IP) FA(Foreign Agent) ERR REQUESTED LIFETIME TOO LONG, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because requested lifetime is too long

- 70 - MIP(Mobile IP) FA(Foreign Agent) ERR MALFORMED REQUEST, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to malformed request
- 71 - MIP(Mobile IP) FA(Foreign Agent) ERR MALFORMED REPLY, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to malformed reply
- 72 - MIP(Mobile IP) FA(Foreign Agent) ERR ENCAPSULATION UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because requested encapsulation is unavailable
- 73 - MIP(Mobile IP) FA(Foreign Agent) ERR VJHC UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because VJ Header Compression is unavailable
- 74 - MIP(Mobile IP) FA(Foreign Agent) ERR REVERSE TUNNEL UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because reverse tunnel is unavailable
- 75 - MIP(Mobile IP) FA(Foreign Agent) ERR REVERSE TUNNEL IS MANDATORY AND T BIT NOT SET, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because reverse tunnel is mandatory but not requested by device
- 79 - MIP(Mobile IP) FA(Foreign Agent) ERR DELIVERY STYLE NOT SUPPORTED, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration because delivery style is not supported
- 97 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING NAI, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing NAI
- 98 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING HA, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing Home Agent
- 99 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING HOME ADDR, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing Home Address
- 104 - MIP(Mobile IP) FA(Foreign Agent) ERR UNKNOWN CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to unknown challenge
- 105 - MIP(Mobile IP) FA(Foreign Agent) ERR MISSING CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to missing challenge
- 106 - MIP(Mobile IP) FA(Foreign Agent) ERR STALE CHALLENGE, this error code is returned when the data call bring up fails in MIP setup phase since foreign agent rejected MIP registration due to stale challenge
- 128 - MIP(Mobile IP) FA(Home Agent) ERR REASON UNSPECIFIED, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration with unspecified reason
- 129 - MIP(Mobile IP) FA(Home Agent) ERR ADMINISTRATIVELY PROHIBITED, this error code is returned when the data call bring up fails in MIP setup phase since home agent administratively prohibited MIP registration
- 130 - MIP(Mobile IP) FA(Home Agent) ERR INSUFFICIENT RESOURCES, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to insufficient resources
- 131 - MIP(Mobile IP) FA(Home Agent) ERR MOBILE NODE AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since home agent fails authentication because MN-HA authenticator was wrong

- 132 - MIP(Mobile IP) FA(Home Agent) ERR FA AUTHENTICATION FAILURE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to FA authentication failure
- 133 - MIP(Mobile IP) FA(Home Agent) ERR REGISTRATION ID MISMATCH, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to registration id mismatch
- 134 - MIP(Mobile IP) FA(Home Agent) ERR MALFORMED REQUEST, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to malformed request
- 136 - MIP(Mobile IP) FA(Home Agent) ERR UNKNOWN HA ADDR, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to unknown Home Agent address. This code is returned by a home agent when the mobile node is performing dynamic home agent address resolution as described in RFC 3220 (IP Mobility Support for IPV4) Sections 3.6.1.1 and 3.6.1.2
- 137 - MIP(Mobile IP) FA(Home Agent) ERR REVERSE TUNNEL UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration because reverse tunnel is unavailable
- 138 - MIP(Mobile IP) FA(Home Agent) ERR REVERSE TUNNEL IS MANDATORY AND T BIT NOT SET, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration because reverse tunnel is mandatory but not requested by device
- 139 - MIP(Mobile IP) FA(Home Agent) ERR ENCAPSULATION UNAVAILABLE, this error code is returned when the data call bring up fails in MIP setup phase since home agent rejected MIP registration due to encapsulation unavailable
- 65536 - MIP ERR REASON UNKNOWN

9.12.2.7 Internal call end reasons (Type=2)

- 200 - INTERNAL MIN, internal error table offset value, no meaningful message to the error.
- 201 - INTERNAL ERROR, this error code is returned when data call is brought down due to some unspecified internal error
- 202 - CALL ENDED
- 203 - INTERNAL UNKNOWN CAUSE CODE, this error code is returned when data call is brought down due to some unknown error
- 204 - UNKNOWN CAUSE CODE, this error code is returned when data call is brought down due to some unknown error
- 205 - CLOSE IN PROGRESS
- 206 - NETWORK INITIATED TERMINATION
- 207 - APP PREEMPTED
- 208 - ERR PDN IPV4 CALL DISALLOWED, this error code is returned when V4 PDN is in throttled state due to network providing only V6 address during the previous VSNCP bring up (subs_limited_to_v6). The time for which the IPv4 PDN is throttled is determined by the IPv4 throttling timers maintained in the profile
- 209 - ERR PDN IPV4 CALL THROTTLED, this error code is returned when V4 PDN is in throttled state due to previous VSNCP bring up failure(s). The time for which the IPv4 PDN is throttled is determined by the IPv4 throttling timers maintained in the profile
- 210 - ERR PDN IPV6 CALL DISALLOWED, this error code is returned when V6 PDN is in throttled state due to network providing only V4 address during the previous VSNCP bring up (subs_limited_to_v4). The time for which the IPv6 PDN is throttled is determined by the IPv6 throttling timers maintained in the profile

- 211 - ERR PDN IPV6 CALL THROTTLED, this error code is returned when V6 PDN is in throttled state due to previous VSNCP bring up failure(s). The time for which the IPv6 PDN is throttled is determined by the IPv6 throttling timers maintained in the profile
- 212 - MODEM RESTART
- 213 - PDP PPP NOT SUPPORTED
- 214 - UNPREFERRED RAT, this error code is returned when data call is brought down since the RAT on which the data call is attempted/connected is no longer the preferred RAT
- 215 - PHYS LINK CLOSE IN PROGRESS, this error code is returned when data call bring up is rejected because physical link is in the process of cleanup
- 216 - APN PENDING HANDOVER, this error code is returned when interface bring up is attempted for an APN that is yet to be handed over to target RAT
- 217 - PROFILE BEARER INCOMPATIBLE
- 218 - MMGSDI CARD EVT, this error code is returned when data call is brought down because card got refreshed/removed
- 219 - LPM OR PWR DOWN, this error code is returned when data call is brought down because device is going into lower power mode or powering down
- 220 - APN DISABLED, this error code is returned when APN is disabled in card
- 221 - MPIT EXPIRED, this error code is returned when data call is brought down because maximum PPP inactivity timer expired
- 222 - IPV6 ADDR TRANSFER FAILED
- 223 - TRAT SWAP FAILED
- 224 - EHRPD TO HRPD FALLBACK, this error code is returned when data call is brought down because device falls back from eHRPD to HRPD (not because of OOS on eHRPD but due to operator/spec driven eHRPD to HRPD fallback requirements)
- 225 - MANDATORY APN DISABLED, this error code is returned when any mandatory APN is disabled, and MinApnList Disallow call config item is set to TRUE in device
- 226 - MIP CONFIG FAILURE, this error code is returned when UE is in MIP Only config (QCMIP=2) but MIP config fails on call bring up due to incorrect provisioning
- 227 - INTERNAL_PDN_INACTIVITY_TIMER_EXPIRED, this error code is returned when PDN inactivity timer expired due to no data transmission in a configurable duration of time
- 228 - MAX_V4_CONNECTIONS, this error code is returned when IPv4 data call bring up is rejected because the UE already maintains the allotted maximum number of IPv4 data connections
- 229 - MAX_V6_CONNECTIONS, this error code is returned when IPv6 data call bring up is rejected because the UE already maintains the allotted maximum number of IPv6 data connections
- 230 - APN_MISMATCH, this error code is returned when New PDN bring up is rejected during interface selection because the UE has already allotted the available interfaces for other PDNs
- 231 - IP_VERSION_MISMATCH, this error code is returned when New call bring up is rejected because the existing data call IP type does not match the requested IP type
- 232 - DUN_CALL_DISALLOWED, this error code is returned when DUN call bring up is rejected because the UE is in eHRPD RAT
- 233 - INVALID_PROFILE, this error code is returned when Call bring up was requested with an invalid profile
- 234 - INTERNAL_EPC_NONEPC_TRANSITION, this error code is returned when Data call is rejected or brought down because the UE is in transition between EPC and non-EPC RAT

- 235 - INVALID_PROFILE_ID, this error code is returned when Call bring up was requested with an invalid profile ID
- 236 - INTERNAL_CALL_ALREADY_PRESENT, this error code is returned when a data call with the same policy is already connected or in the process of bring up on another RmNet instance
- 237 - IFACE_IN_USE, this error code is returned when the current IFACE is in use
- 238 - IP_PDP_MISMATCH, this error code is returned when a PPP call is attempted on a PDP profile
- 239 - APN_DISALLOWED_ON_ROAMING, this error code is returned when PDN connection to the APN is disallowed on the roaming network. The DS generates DS_SYS_EVENT_3GPP_ROAMING_DISALLOWED_INFO to inform clients of the APNs that must deregister and disconnect (if the PDN is up) while the UE is on the roaming network. Clients must deregister or disconnect if the APN name corresponding to the client is listed in this system event. Clients can register again if the APN name corresponding to the client is not listed in this system event.
- 240 - APN_PARAM_CHANGE, this error code is returned when Failure to reestablish the PDN with the changed parameters; when APN-related parameters are changed, the PDN associated with the parameters must be brought down and reestablished with the changed parameters
- 241 - IFACE_IN_USE_CFG_MATCH, this error code is returned when the IFACE is already in use with a matching configuration
- 242 - NULL_APN_DISALLOWED, this error code is returned when a PDN is attempted to be brought up with a NULL APN and when a NULL APN is not supported
- 243 - THERMAL_MITIGATION, this error code is returned when the thermal level increases and causes calls to be torn down when normal mode of operation is not allowed
- 244 - SUBS_ID_MISMATCH, this error code is returned when a new call bring up is rejected due to a mismatch between the subs_id in the profile and the subs_id in the ACL policy information
- 245 - DATA_SETTINGS_DISABLED, this error code is returned when the PDN connection to a given APN is disallowed because data is disabled from the device user interface (UI) settings
- 246 - DATA_ROAMING_SETTINGS_DISABLED, this error code is returned when the PDN Connection to a given APN is disallowed because data roaming is disabled from the device UI settings and the UE is roaming
- 247 - APN_FORMAT_INVALID, this error code is returned when the APN specified in the policy does not follow the format specified in the 3GPP Specification
- 248 - DDS_CALL_ABORT, this error code is returned when a DDS switch occurs; the data call is brought down with the DDS abort reason
- 249 - VALIDATION_FAILURE, this error code is returned when a data call is brought down due to an internal validation failure

9.12.2.8 Call Manager defined call end reasons (Type=3)

- 500 - CDMA LOCK, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to device in CDMA locked state
- 501 - INTERCEPT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since it received an intercept order from the base station
- 502 - REORDER, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to receiving a reorder from base station
- 503 - REL SO REJ, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to receiving a release from base station with reason: SO Reject
- 504 - INCOM CALL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since it received an incoming call from base station

- 505 - ALERT STOP, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to RL/FL fade (or) receiving call release from base stations
- 506 - ACTIVATION, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to channel acquisition failures. This indicates that device has failed acquiring all the channels in the PRL
- 507 - MAX ACCESS PROBE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due maximum access probes transmitted
- 508 - CCS NOT SUPPORTED BY BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since concurrent service is not supported by base station
- 509 - NO RESPONSE FROM BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since there is no response received from base station
- 510 - REJECTED BY BS, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to base station rejecting the call
- 511 - INCOMPATIBLE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since concurrent services requested were not compatible
- 512 - ALREADY IN TC, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since traffic channel is already up for voice calls
- 513 - USER CAL ORIG DURING GPS
- 514 - USER CAL ORIG DURING SMS, this error code is returned when data call is brought down because traffic channel request got rejected since SMS is ongoing
- 515 - NO CDMA SRV, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device does not have CDMA service
- 516 - MC ABORT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since MC aborted the origination/conversation
- 517 - PSIST NG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to persistence test failure
- 518 - UIM NOT PRESENT, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to RUIM not present
- 519 - RETRY ORDER, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due to receiving a retry order from base station
- 520 - ACCESS BLOCK, this error code is returned when data call is brought down because traffic channel rejected/released due to Access blocked by base station
- 521 - ACCESS BLOCK ALL, this error code is returned when data call is brought down because traffic channel rejected due to Access blocked by the base station for all mobile devices
- 522 - IS707B MAX ACC, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) due maximum access probes for IS-707B call
- 523 - THERMAL EMERGENCY, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) to put device in thermal emergency
- 524 - CALL ORIG THROTTLED, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since call origination is throttled by DCTM module
- 535 - USER CALL ORIGINATE DURING VOICE CALL, this error code is returned when data call is brought down because traffic channel got released by CM(Call Manager) in favor of voice call or SMS when concurrent voice and data are not supported
- 1000 - CONFERENCE FAILED

- 1001 - INCOMING REJECTED
- 1002 - NO GATEWAY SERVICE
- 1003 - NO GPRS CONTEXT
- 1004 - ILLEGAL MS, This cause is sent to the MS when the network refuses service to the MS either because an identity of the MS is not acceptable to the network or because the MS does not pass the authentication check, i.e. the SRES received from the MS is different from that generated by the network
- 1005 - ILLEGAL ME, This cause is sent to the MS if the ME used is not acceptable to the network, e.g. blacklisted
- 1006 - GPRS SERVICES AND NON GPRS SERVICES NOT ALLOWED
- 1007 - GPRS SERVICES NOT ALLOWED
- 1008 - MS IDENTITY CANNOT BE DERIVED BY THE NETWORK
- 1009 - IMPLICITLY DETACHED, this error code is sent to the MS either if the network has implicitly detached the MS, e.g. some while after the Mobile reachable timer has expired, or if the GMM context data related to the subscription does not exist in the SGSN e.g. because of a SGSN restart.
- 1010 - PLMN NOT ALLOWED, this error code is sent to the MS if it requests location updating in a PLMN where the MS, by subscription or due to operator determined barring is not allowed to operate
- 1011 - LOCAL AREA NOT ALLOWED
- 1012 - GPRS SERVICES NOT ALLOWED IN THIS PLMN
- 1013 - PDP DUPLICATE
- 1014 - USER EQUIPMENT RADIO ACCESS TECHNOLOGY CHANGE
- 1015 - CONGESTION
- 1016 - NO PDP CONEXT ACTIVATED
- 1017 - ACCESS CLASS DSAC REJECTION
- 1018 - PDP ACTIVATE MAX RETRY FAILED
- 1019 - RAB FAILURE
- 1020 - EPS SERVICE NOT ALLOWED
- 1021 - TRACKING AREA NOT ALLOWED
- 1022 - ROAMING NOT ALLOWED IN THIS TRACKING AREA
- 1023 - NO SUITABLE CELLS IN TRACKING AREA
- 1024 - NOT AUTHORIZED FOR THIS CLOSED SUBSCRIBER GROUP
- 1025 - ESM UNKNOWN EPS BEARER CONTEXT
- 1026 - DRB RELEASED AT RRC
- 1027 - NAS SIG CONN RELEASED
- 1028 - EPS MOBILITY MANAGEMENT DETACHED
- 1029 - EPS MOBILITY MANAGEMENT ATTACH FAILED
- 1030 - EPS MOBILITY MANAGEMENT ATTACH STARTED
- 1031 - LTE NAS SERVICE REQ FAILED
- 1032 - ESM(EPS Session Management) ACTIVE DEDICATED BEARER REACTIVATED BY NW

- 1033 - ESM(EPS Session Management) LOWER LAYER FAILURE
- 1034 - ESM(EPS Session Management) SYNC UP WITH NW
- 1035 - ESM(EPS Session Management) NW ACTIVATED DED BEARER WITH ID OF DEF BEARER
- 1036 - ESM(EPS Session Management) BAD OTA MESSAGE
- 1037 - ESM DS REJECTED THE CALL
- 1038 - ESM(EPS Session Management) CONTEXT TRANSFERRED DUE TO IRAT
- 1039 - DS EXPLICIT DEACT
- 1040 - ESM(EPS Session Management) LOCAL CAUSE NONE
- 1041 - LTE NAS SERVICE REQ FAILED NO THROTTLE
- 1042 - ACL FAILURE, This error code should rarely triggered and reported to the application
- 1043 - LTE NAS SERVICE REQ FAILED DS DISALLOW
- 1044 - EMM(EPS Mobility Management) T3417 EXPIRED
- 1045 - EMM(EPS Mobility Management) T3417 EXT EXPIRED
- 1046 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE TXN - Light Radio Resource Controller Uplink data confirmation failure
- 1047 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE HO
- 1048 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE CONN REL
- 1049 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE RLF
- 1050 - LRRC(LTE Radio Resource Control) UL DATA CNF FAILURE CTRL NOT CONN
- 1051 - LRRC(LTE Radio Resource Control) CONN EST FAILURE
- 1052 - LRRC(LTE Radio Resource Control) CONN EST FAILURE ABORTED
- 1053 - LRRC(LTE Radio Resource Control) CONN EST FAILURE ACCESS BARRED
- 1054 - LRRC(LTE Radio Resource Control) CONN EST FAILURE CELL RESEL
- 1055 - LRRC(LTE Radio Resource Control) CONN EST FAILURE CONFIG FAILURE
- 1056 - LRRC(LTE Radio Resource Control) CONN EST FAILURE TIMER EXPIRED
- 1057 - LRRC(LTE Radio Resource Control) CONN EST FAILURE LINK FAILURE
- 1058 - LRRC(LTE Radio Resource Control) CONN EST FAILURE NOT CAMPED
- 1059 - LRRC(LTE Radio Resource Control) CONN EST FAILURE SI FAILURE
- 1060 - LRRC(LTE Radio Resource Control) CONN EST FAILURE CONN REJECT
- 1061 - LRRC(LTE Radio Resource Control) CONN REL NORMAL
- 1062 - LRRC(LTE Radio Resource Control) CONN REL RLF
- 1063 - LRRC(LTE Radio Resource Control) CONN REL CRE FAILURE
- 1064 - LRRC(LTE Radio Resource Control) CONN REL OOS DURING CRE
- 1065 - LRRC(LTE Radio Resource Control) CONN REL ABORTED
- 1066 - LRRC(LTE Radio Resource Control) CONN REL SIB READ ERROR
- 1067 - DETACH WITH REATTACH LTE NW DETACH

- 1068 - DETACH WITH OUT REATTACH LTE NW DETACH
- 1069 - ESM(EPS Session Management) PROC TIME OUT
- 1070 - MESSAGE EXCEED MAX L2 LIMIT
- 1500 - CD GEN OR BUSY, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to the reception of a Connection Deny message with a deny code of general or network busy
- 1501 - CD BILL OR AUTH, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to the reception of a Connection Deny message with a deny code of billing failure or authentication failure
- 1502 - CHG HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since there is a change to HDR system due to redirection or PRL not preferred
- 1503 - EXIT HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device exited HDR due to redirection or PRL not preferred
- 1504 - HDR NO SESSION, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device does not have a HDR session
- 1505 - HDR ORIG DURING GPS FIX, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since it is ending an HDR call origination in favor of a GPS fix
- 1506 - HDR CS TIMEOUT, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since connection setup on HDR system timed out
- 1507 - HDR RELEASED BY CM, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) when it wants to release a HDR call so a 1X call can continue
- 1508 - COLLOC ACQ FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) when device failed to acquire co-located HDR for origination
- 1509 - OTASP COMMIT IN PROG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since an OTASP commit is in progress
- 1510 - NO HYBR HDR SRV, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device has no Hybrid HDR service
- 1511 - HDR NO LOCK GRANTED, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since HDR module could not get the RF lock
- 1512 - HOLD OTHER IN PROG, this error code is returned when data call is brought down by CM(Call Manager) because DBM or SMS is in progress
- 1513 - HDR FADE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since HDR module released the call due to fade
- 1514 - HDR ACC FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to HDR system Access Failure
- 2000 - CLIENT END, this error code is returned when client ends the data call
- 2001 - NO SRV, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since device has no service
- 2002 - FADE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device lost the system due to fade
- 2003 - REL NORMAL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to receiving a release from base station with no reason

- 2004 - ACC IN PROG, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to Access attempt already in progress
- 2005 - ACC FAIL, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) due to Access Failure
- 2006 - REDIR OR HANDOFF, this error code is returned when data call is brought down because device is in the process of redirecting/handing off to a different target system
- 2500 - OFFLINE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device went offline
- 2501 - EMERGENCY MODE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since device is operating in Emergency mode
- 2502 - PHONE IN USE, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since device is in use (e.g voice call)
- 2503 - INVALID MODE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since the device's operational mode is different from the mode requested in the traffic channel bring up
- 2504 - INVALID SIM STATE, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since the SIM was marked by network as invalid for circuit and/or packet service domain
- 2505 - NO COLLOC HDR, this error code is returned when data call is brought down because traffic channel got rejected/released by CM(Call Manager) since there is no collocated HDR
- 2506 - CALL CONTROL REJECTED, this error code is returned when data call is brought down because traffic channel request got rejected by CM(Call Manager) since Call control module rejected the request

9.12.2.9 3GPP specification defined call end reasons (Type=6)

- 8 - OPERATOR DETERMINED BARRING, this reason code is posted by the MME(Mobility Management Entity) to indicate operator has barred the UE
- 25 - LLC SNDCP FAILURE, PDP context deactivation initiated by the MS or by the Network
- 26 - INSUFFICIENT RESOURCES, this reason is posted to indicate that the network cannot provide the requested service due to insufficient resources
- 27 - MISSING OR UNKNOWN APN, the APN was required and not specified or APN could not be resolved. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may endup throttling further requests to the same APN
- 28 - UNKNOWN PDN TYPE, the reason is posted by the network to indicate that the PDN type was not recognized
- 29 - AUTH FAILED, the reason is posted when authentication fails. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may end up throttling further requests to the same APN
- 30 - GGSN REJECT, the reason is posted when the request was rejected by Serving GW or PDN GW. In LTE mode of operation, this is a PDN throttling cause code, meaning the UE may end up throttling further requests to the same APN
- 31 - ACTIVATION REJECT, the reason is posted when the request is rejected by the network due to unspecified reasons
- 32 - OPTION NOT SUPPORTED, the reason is posted when UE requested a service not supported by the PLMN
- 33 - OPTION UNSUBSCRIBED, This cause is sent when the MS requests a service option for which it has no subscription

- 34 - OPTION TEMP OOO, service option temporarily out of order, this reason is posted when the network is temporarily out of resources to service the request
- 35 - PTI ALREADY USED, the reason is posted to indicate that PTI (Procedure Transaction Identifier) used in the request is already active via another UE requested procedure
- 36 - REGULAR DEACTIVATION, this reason is posted by the network to initiate a regular release of bearer resources
- 37 - EPS QOS NOT ACCEPTED, this reason is posted by the network to indicate that the QoS requested by the UE could not be accepted
- 38 - NETWORK FAILURE, this reason is posted when an error occurs in the network
- 39 - UMTS REACTIVATION REQ, this reason is posted by the network to request for bearer reactivation. This code may be posted during network congestion
- 40 - FEATURE NOT SUPPORTED, Unsuccessful MBMS context activation requested by the network
- 41 - TFT SEMANTIC ERROR, the reason is posted by the network to indicate semantic error(s) in specifying TFT operation included in the request
- 42 - TFT SYNTAX ERROR, the reason is posted by the network to indicate syntactic error(s) in specifying TFT operation included in the request
- 43 - UNKNOWN PDP CONTEXT, the reason is posted when the bearer identity (or linked bearer identity) in the request is invalid (or inactive)
- 44 - FILTER SEMANTIC ERROR, the reason is posted by the network to indicate semantic error(s) in specifying packet filter(s) associated with a TFT
- 45 - FILTER SYNTAX ERROR, the reason is posted by the network to indicate syntactic error(s) in specifying packet filter(s) associated with a TFT
- 46 - PDP WITHOUT ACTIVE TFT, the reason is posted by the network when UW requested more than one PDP connection without TFT
- 50 - IPV4 ONLY ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 50, PDN type IPv4 only allowed.
- 51 - IPV6 ONLY ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 51, PDN type IPv6 only allowed
- 52 - SINGLE ADDRESS BEARER ONLY, 3GPP equivalent ESM(EPS Session Management) cause code value 52, Single address bearers only allowed. The reason is posted when the network supports single address bearers only, meaning dual IP bearers are not supported
- 53 - ESM INFORMATION NOT RECEIVED, 3GPP equivalent ESM(EPS Session Management) cause code value 53, ESM information not received. The reason is posted by the network to indicate that the PDN connection request was rejected because ESM information was not received
- 54 - PND CONNECTION DOES NOT EXIST, 3GPP equivalent ESM(EPS Session Management) cause code value 54, PDN connection does not exist The reason is posted by the network during handover from a non-3GPP network to indicate that the MME does not have any information regarding the requested PDN connection
- 55 - MULTIPLE CONNECTION TO SAME PDN NOT ALLOWED, 3GPP equivalent ESM(EPS Session Management) cause code value 55, Multiple PDN connections for a given APN not allowed. The reason is posted by the network to indicate that the UE is already connected to the requested APN via another PDN/PDN connection
- 81 - INVALID TRANSACTION ID, the reason is posted by the network to indicate that the PTI used in the request is unassigned or reserved
- 95 - MESSAGE INCORRECT SEMANTIC, the reason is posted by the network to indicate receipt of an invalid message

- 96 - INVALID MANDATORY INFO, the reason is posted by the network to indicate receipt of a message with semantic error in a mandatory information element
- 97 - MESSAGE TYPE UNSUPPORTED, the reason is posted by the network to indicate receipt of a message that is either undefined or defined but not implemented by the equipment sending this ESM cause
- 98 - MSG TYPE NONCOMPATIBLE STATE, the reason is posted by the network to indicate receipt of a message type that cannot be handled in the current network protocol state
- 99 - UNKNOWN INFO ELEMENT, the reason is posted by the network to indicate receipt of a message that includes and information element that is either not defined or defined but not implemented by the equipment sending the ESM cause
- 100 - CONDITIONAL IE ERROR, the reason is posted by the network to indicate receipt of a message that includes a syntactically incorrect information element. This message is ignored by the network.
- 101 - MSG AND PROTOCOL STATE UNCOMPATIBLE, the reason is posted by the network to indicate receipt of a message that cannot be handled in the current network protocol state
- 111 - PROTOCOL ERROR, the reason is posted by the network to indicate a protocol error when no other error applies
- 112 - APN TYPE CONFLICT
- 113 - INVALID PROXY-CALL SESSION CONTROL FUNCTION ADDRESS

9.12.2.10 PPP call end reasons (Type=7)

- 1 - TIMEOUT, this error code is returned when the data call bring up fails in PPP setup due to timeout (For e.g: LCP Conf Ack not received from network)
- 2 - AUTH FAILURE, this error code is returned when the data call bring up fails in PPP setup due to authentication failure
- 3 - OPTION MISMATCH, this error code is returned when the data call bring up fails in PPP setup due option mismatch (e.g: Authentication is required, but not negotiated with network during LCP phase)
- 31 - PAP FAILURE, this error code is returned when the data call bring up fails in PPP setup due to PAP failure
- 32 - CHAP FAILURE, this error code is returned when the data call bring up fails in PPP setup due to CHAP failure
- 33 - CLOSE IN PROGRESS, this error code is returned when the data call bring up fails in PPP setup since PPP is in the process of cleaning the previous PPP session
- -1 - UNKNOWN, this error code is unused

9.12.2.11 EHRPD call end reasons (Type=8)

- 1 - SUBS LIMITED TO V4, this error code is returned when the V6 interface bring up fails because network provided only V4 address for the upcoming PDN
- 2 - SUBS LIMITED TO V6, this error code is returned when the V4 interface bring up fails because network provided only V6 address for the upcoming PDN
- 4 - VSNCP(Vendor Specific Network Control Protocol) TIMEOUT, this error code is returned when the data call bring up fails in VSNCP phase due to VSNCP timeout error
- 5 - VSNCP(Vendor Specific Network Control Protocol) FAILURE, this error code is returned when VSNCP configuration failed during call bring up
- 6 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I GEN ERROR, this error code is returned when the data call bring up fails in VSNCP phase due to general error

- 7 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I UNAUTH APN, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason requested APN is unauthorized
- 8 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN LIMIT EXCEED, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN limit exceeded
- 9 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I NO PDN GW, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason no PDN gateway
- 10 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN GW UNREACH, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN gateway unreachable
- 11 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN GW REJ, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN gateway reject
- 12 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I INSUFF PARAM, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason insufficient parameter
- 13 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I RESOURCE UNAVAIL, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason resource unavailable
- 14 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I ADMIN PROHIBIT, this error code is returned when the data call bring up fails in SNCP phase since network rejected VSNCP config request with reason admin prohibited
- 15 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN ID IN USE, this error code is returned when the data call bring up fails in VSNCP phase because network rejected with reason PDN ID IN USE (or) All existing PDNs are brought down with this end reason because one of the PDN bring up got rejected by network with reason PDN ID IN USE
- 16 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I SUBSCR LIMITATION, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason subscriber limitation
- 17 - VSNCP(Vendor Specific Network Control Protocol) 3GPP2I PDN EXISTS FOR THIS APN, this error code is returned when the data call bring up fails in VSNCP phase since network rejected VSNCP config request with reason PDN exists for this APN

9.12.2.12 IPV6 call end reasons (Type=9)

- 1 - PREFIX UNAVAILABLE, this error code is returned when V6 data call is brought down because device failed to get the prefix from network
- 2 - IPV6 ERR HRPD IPV6 DISABLED, this error code is returned when V6 data call bring up is rejected because IPV6 is disabled in 1X/HRPD mode
- 3 - IPV6 DISABLED, this error code is returned when IPv6 data call bring up is rejected because NV1896 (IPV6 enable) is disabled

Copyright: © 2011-2013 Sierra Wireless, Inc. all rights reserved

9.13 qaGobiApiTableCarrierCodes.h File Reference

Carrier Codes table.

9.13.1 Detailed Description

Carrier Codes table.

9.13.2 Carrier Codes (Number - Carrier)

- 0 - no carrier specified
- 1 - Generic
- 2 - Telstra
- 4 - AT&T
- 5 - Verizon
- 11 - Sprint
- 12 - Telefonica
- 101 - Verizon
- 102 - Sprint
- 103 - Alltel
- 104 - Bell Mobility
- 105 - Telus
- 106 - U.S. Cellular
- 107 - Telstra
- 108 - China Unicom
- 109 - Telecom New Zealand
- 110 - SK Telecom
- 111 - Reliance Communications
- 112 - Tata Communications
- 113 - MetroPCS Communications
- 114 - Leap Wireless
- 115 - KDDI
- 116 - Grupo Iusacell
- 117 - China Telecom
- 118 - Open Mobile Handset
- 176 - Rogers
- 177 - NetIndex
- 178 - DNA
- 179 - Big Pond
- 201 - AT&T
- 202 - Vodafone

- 203 - T-Mobile
- 204 - Orange
- 205 - Telefonica
- 206 - Telecom Italia
- 207 - 3
- 208 - O2
- 209 - SFR
- 210 - Swisscom AG
- 211 - China Mobile
- 212 - Telstra
- 213 - Singapore Telecommunications
- 214 - Reliance Telecommunications
- 215 - Bharti Airtel
- 216 - NTT docomo
- 217 - E Mobile
- 218 - Softbank
- 219 - Korea Telecom Freetel
- 220 - SK Telecom
- 221 - Telenor
- 222 - NetCom Norway
- 223 - TeliaSonera
- 224 - América Móvil
- 225 - Brasil Vivo
- 0xFFFFFFFF - Unknown

Copyright: © 2011-2014 Sierra Wireless, Inc. all rights reserved

9.14 qaGobiApiTableCodingScheme.h File Reference

Data Coding Scheme.

Macros

- #define __GOBI_API_CODING_SCHEME_H__

9.14.1 Detailed Description

Data Coding Scheme.

9.14.2 Call Control Result Reasons (Value - Name - Description)

9.14.2.1 Use of bits 3..0\n\n

- Language using the GSM 7 bit default alphabet Bits 3..0
  indicate the language:
  0000 German
  0001 English
  0010 Italian
  0011 French
  0100 Spanish
  0101 Dutch
  0110 Swedish
  0111 Danish
  1000 Portuguese
  1001 Finnish
  1010 Norwegian
  1011 Greek
  1100 Turkish
  1101 Hungarian
  1110 Polish
  1111 Language unspecified

9.14.3 Coding Group Bits 7..4(0001)

9.14.3.1 use of bits 3..0

- 0000 GSM 7 bit default alphabet; message preceded by language indication.
  The first 3 characters of the message are a two-character representation of the language encoded according to ISO 639 [12], followed by a CR character. The CR character is then followed by 90 characters of text.
- 0001 UCS2; message preceded by language indication
  The message starts with a two GSM 7-bit default alphabet character representation of the language encoded according to ISO 639. This is padded to the octet boundary with two bits set to 0 and then followed by 40 characters of UCS2-encoded message.
  An MS not supporting UCS2 coding will present the two character language identifier followed by improperly interpreted user data.

9.14.4 Coding Group Bits 7..4(0010)

9.14.4.1 use of bits 3..0

- 0000 Czech
  0001 Hebrew
  0010 Arabic

0011 Russian
0100 Icelandic
0101..1111 Reserved for other languages using the GSM 7 bit
default alphabet, with unspecified handling at the MS

9.14.5 Coding Group Bits 7..4(0011)

9.14.5.1 use of bits 3..0

- 0000..1111 Reserved for other languages using the GSM 7 bit
  default alphabet, with unspecified handling at the MS

9.14.6 Coding Group Bits 7..4(01xx)

9.14.6.1 use of bits 3..0

- General Data Coding indication
  - Bits 5..0 indicate the following:
  - Bit 5, if set to 0, indicates the text is uncompressed
  - Bit 5, if set to 1, indicates the text is compressed using the compression algorithm defined in 3GPP TS 23.042
  - Bit 4, if set to 0, indicates that bits 1 to 0 are reserved and have no message class meaning
  - Bit 4, if set to 1, indicates that bits 1 to 0 have a message class meaning: Bit 1 Bit 0 Message Class:
    0 0 Class 0
    0 1 Class 1 Default meaning: ME-specific.
    1 0 Class 2 (U)SIM specific message.
    1 1 Class 3 Default meaning: TE-specific (see 3GPP TS 27.005)
    Bits 3 and 2 indicate the character set being used, as follows:
    Bit 3 Bit 2 Character set:
    0 0 GSM 7 bit default alphabet
    0 1 8 bit data
    1 0 UCS2 (16 bit) [10]
    1 1 Reserved

9.14.7 Coding Group Bits 7..4(1001)

9.14.7.1 Reserved coding groups

- Message with User Data Header (UDH) structure:
  Bit 1 Bit 0 Message Class:
  0 0 Class 0
  0 1 Class 1 Default meaning: ME-specific.
  1 0 Class 2 (U)SIM specific message.
  1 1 Class 3 Default meaning: TE-specific (see 3GPP TS 27.005 [8])
  Bits 3 and 2 indicate the alphabet being used, as follows:
  Bit 3 Bit 2 Alphabet:
  0 0 GSM 7 bit default alphabet
  0 1 8 bit data
  1 0 USC2 (16 bit) [10]
  1 1 Reserved

9.14.8 Coding Group Bits 7..4(1010..1101)

9.14.8.1 Reserved coding groups

9.14.9 Coding Group Bits 7..4(1110)

9.14.9.1 Defined by the WAP Forum

9.14.10 Coding Group Bits 7..4 (1111)

9.14.10.1 Data coding / message handling

- Bit 3 is reserved, set to 0.

  Bit 2 Message coding:

  0 GSM 7 bit default alphabet

  1 8 bit data

  Bit 1 Bit 0 Message Class:

  0 0 No message class.

  0 1 Class 1 user defined.

  1 0 Class 2 user defined.

  1 1 Class 3

  default meaning: TE specific(3GPP TS 27.005)

  Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.14.11 Macro Definition Documentation

9.14.11.1 #define __GOBI_API_CODING_SCHEME_H__

9.15 qaGobiApiTableGpsCapabilityCodes.h File Reference

Position Determination Service API GPS Capability Codes.

9.15.1 Detailed Description

Position Determination Service API GPS Capability Codes.

9.15.2 GPS capability (Value - Capability)

- 0 - None
- 1 - Standalone
- 2 - Assisted (including XTRA and implying standalone is also supported)
- 3 - Assisted (without XTRA and implying standalone is also supported)
- 0xFFFFFFFF - Unknown

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.16 qaGobiApiTablePowerModes.h File Reference

Device Management Service API Power Modes table.

9.16.1 Detailed Description

Device Management Service API Power Modes table.

9.16.2 Power Modes (Value - Description)

- 0 - Online (default)
- 1 - Low power (airplane) mode
- 2 - Factory test mode
- 3 - Offline
- 4 - Reset
- 5 - Power off
- 6 - Persistent low power (airplane) mode
- 7 - Mode - only low power

Valid transitions for Power Modes

- Online to Low Power, Persistent low power, Factory test, Offline or Shut Down
- Low power to online, Persistent low power, Offline, or Shut Down
- Persistent low power to Online, Low power, Offline or Shut down
- Factory test to online
- Offline to Reset

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.17 qaGobiApiTableRadioInterfaces.h File Reference

Network Access Service API Radio Interfaces table.

9.17.1 Detailed Description

Network Access Service API Radio Interfaces table.

9.17.2 Radio interface

9.17.2.1 Technology (Value - Radio Interface Technology)

- 0 - No service
- 1 - CDMA 1xRTT
- 2 - CDMA 1xEV-DO

- 3 - AMPS (Unsupported)
- 4 - GSM
- 5 - UMTS
- 6 - WLAN
- 7 - GPS
- 8 - LTE

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.18 qaGobiApiTableRegionCodes.h File Reference

Region Codes table.

9.18.1 Detailed Description

Region Codes table.

9.18.2 Region Codes (Code - Region)

- 0 - North America
- 1 - Latin America
- 2 - Europe
- 3 - Asia
- 4 - Australia
- 5 - Global
- 0xFFFFFFFF - Unknown

Copyright: © 2011 Sierra Wireless, Inc. all rights reserved

9.19 qaGobiApiTableServiceOptions.h File Reference

Voice Service Options.

9.19.1 Detailed Description

Voice Service Options.

9.19.2 Service Option codes (Code - Reason)

9.19.2.1 Description

- 0x0001 - Basic variable rate voice service (8 kbps)
- 0x0002 - Mobile station loopback (8 kbps)
- 0x0003 - Enhanced variable rate voice service (8 kbps)
- 0x0004 - Asynchronous data service (9.6 kbps)
- 0x0005 - Group 3 facsimile (9.6 kbps)
- 0x0006 - Short message service (rate set 1)
- 0x0007 - Packet data service: Internet or ISO Protocol stack (9.6 kbps)
- 0x0008 - Packet data service: CDPD Protocol stack (9.6 kbps)
- 0x0009 - Mobile station loopback (13 kbps)
- 0x000A - transparent service
- 0x000B - III nontransparent service
- 0x000C - Asynchronous data service (14.4 or 9.6 kbps)
- 0x000D - Group 3 facsimile (14.4 or 9.6 kbps)
- 0x000E - Short message service (rate set 2)
- 0x000F - Packet data service: Internet or ISO Protocol stack (14.4 kbps)
- 0x0010 - Packet data service: CDPD Protocol stack (14.4 kbps)
- 0x0011 - High-rate voice service (13 kbps)
- 0x0012 - Over-the-air parameter administration (rate set 1)
- 0x0013 - Over-the-air parameter administration (rate set 2)
- 0x0014 - Group 3 analog facsimile (rate set 1)
- 0x0015 - Group 3 analog facsimile (rate set 2)
- 0x0016 - High-speed packet data service: Internet or ISO Protocol stack (RS1 forward, RS1 reverse)
- 0x0017 - High-speed packet data service: Internet or ISO Protocol stack (RS1 forward, RS2 reverse)
- 0x0018 - High-speed packet data service: Internet or ISO Protocol stack (RS2 forward, RS1 reverse)
- 0x0019 - High-speed packet data service: Internet or ISO Protocol stack (RS2 forward, RS2 reverse)
- 0x001A - High-speed packet data service: CDPD Protocol stack (RS1 forward, RS1 reverse)
- 0x001B - High-speed packet data service: CDPD Protocol stack (RS1 forward, RS2 reverse)
- 0x001C - High-speed packet data service: CDPD Protocol stack (RS2 forward, RS1 reverse)
- 0x001D - High-speed packet data service: CDPD Protocol stack (RS2 forward, RS2 reverse)
- 0x001E - RATE_SET_1 Supplemental channel loopback test for rate set 1
- 0x001F - RATE_SET_2 Supplemental channel loopback test for rate set 2
- 0x0020 - Test Data Service Option (TDSO)
- 0x0021 - cdma2000 high-speed packet data service, Internet or ISO Protocol stack

- 0x0022 - cdma2000 high-speed packet data service, CDPD Protocol
- 0x0023 - Location services, rate set 1 (9.6 kbps)
- 0x0024 - Location services, rate set 2 (14.4 kbps)
- 0x0025 - ISDN interworking service (64 kbps)
- 0x0026 - GSM voice
- 0x0027 - GSM circuit data
- 0x0028 - GSM packet data
- 0x0029 - GSM short message service
- 0x0036 - Markov Service Option (MSO)
- 0x0037 - Loopback Service Option (LSO)
- 0x0038 - Selectable mode vocoder
- 0x0039 - 32 kbps circuit video conferencing
- 0x003A - CONFERENCING 64 kbps circuit video conferencing
- 0x003B - HRPD packet data service, which when used in paging over the 1X air interface, a page response is not required
- 0x003C - Link Layer Assisted Robust Header Compression (LLA ROHC) - header removal
- 0x003D - LLA ROHC - Header Compression
- 0x003E - Source-controlled Variable-Rate Multimode Wideband (VMR-WB) speech codec rate set 2
- 0x003F - Source-controlled VMR-WB speech codec rate set 1
- 0x0040 - HRPD auxiliary packet data service instance
- 0x0041 - cdma2000/GPRS interworking
- 0x0042 - ISO_PROTOCOL_SO_66 cdma2000 high-speed packet data service, Internet or ISO Protocol stack
- 0x0043 - HRPD packet data IP service where higher layer protocol is IP or ROHC
- 0x0044 - Enhanced variable rate voice service (EVRC-B)
- 0x0045 - HRPD packet data service, which when used in paging over the 1X air interface, a page response is required
- 0x0046 - Enhanced variable rate voice service (EVRC-WB)
- 0x1004 - Asynchronous data service, Revision 1 (9.6 or 14.4 kbps)
- 0x1005 - Group 3 facsimile, Revision 1 (9.6 or 14.4 kbps)
- 0x1007 - Packet data service: Internet or ISO Protocol stack, Revision 1 (9.6 or 14.4 kbps)
- 0x1008 - Packet data service: CDPD Protocol stack, Revision 1 (9.6 or 14.4 kbps)
- 0x7FF8 - Identifies service reference identifier 0
- 0x7FF9 - Identifies service reference identifier 1
- 0x7FFA - Identifies service reference identifier 2
- 0x7FFB - Identifies service reference identifier 3
- 0x7FFC - Identifies service reference identifier 4
- 0x7FFD - Identifies service reference identifier 5

- 0x7FFE - Identifies service reference identifier 6
- 0x7FFF - Identifies service reference identifier 7

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.20 qaGobiApiTableSupServiceInfoClasses.h File Reference

Voice Supplementary Service Information Classes.

9.20.1 Detailed Description

Voice Supplementary Service Information Classes.

9.20.2 Supplementary Service Information Classes (Value - Service Class)

- 0X00 - CLASS_NONE
- 0X01 - CLASS_VOICE
- 0X02 - CLASS_DATA
- 0X04 - CLASS_FAX
- 0X08 - CLASS_SMS
- 0X10 - CLASS_DATACIRCUITSYNC
- 0X20 - CLASS_DATACIRCUITASYNC
- 0X40 - CLASS_PACKETACCESS
- 0X80 - CLASS_PADACCESS

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.21 qaGobiApiTableSwiAudio.h File Reference

Swi Audio related tables.

9.21.1 Detailed Description

Swi Audio related tables.

9.21.2 ACDB Device (Device ID - description)

- 0 - Vehicle HF
- 1 - Handset
- 2 - TTY
- 3 - USB
- 4 - NA

9.21.3 Physical Interface (Device ID - description - Interface parameters)

- 0 - PCM - Mode: 0-slave, 1-master, 2-Auxiliary PCM; Rate: 0-8k, 1-16k; Format: 0-linear, 1-u-law, 2-A-law; Padding: 0-disable, 1-enable; Bits-frame: 0-8BPF, 1-16BPF, 2-32BPF, 3-64BPF, 4-128BPF, 5-256BPF;
- 1 - I2S - None
- 2 - Analog(with internal codec) - None
- 3 - USB - None

Copyright: © 2013 Sierra Wireless, Inc. all rights reserved

9.22 qaGobiApiTableSwiOMADMSessionStatus.h File Reference

Session Status table.

9.22.1 Detailed Description

Session Status table.

9.22.2 OMA DM Session Status (Session Status - Meaning - Usage)

- 0x0000 - Successful - Session succeeded
- 0x0001 - Break - Session succeeded
- 0x0002 - Large Object Handled - Session succeeded
- 0x0003 - No status - Session succeeded
- 0x0004 - No more commands - Session succeeded
- 0x0005 - User cancel - Session cancelled
- 0x0100-0x01FF - General errors
- 0x0200-0x02FF - Syncml errors
- 0x0300-0x03FF - Authentication errors
- 0x0400-0x04FF - Protocol errors
- 0x0500-0x05FF - Tree errors (DM Only)
- 0x0600-0x06FF - Not applicable
- 0x0700-0x07FF - Trigger errors
- 0x0800-0x08FF - FUMO errors
- 0x0900-0x09FF - Communication errors
- 0x0A00-0x0AFF - Parsing errors
- 0x0B00-0x0CFF - Not applicable
- 0x7F00-0x7F12 - Insignia errors
- 0x7F13 - Illegal text - Text received contains illegal characters
- 0x7F14 - Download failure - Failed to download FOTA image

- 0x7F15 - Empty session - Session ran successfully, but no information was updated
- 0x7F16 - Factory reset successful - Factory reset succeeded
- 0x7F17 - Factory reset fail - Factory reset failed

Copyright: © 2017 Sierra Wireless, Inc. all rights reserved

9.23 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference

Update Complete Status table.

9.23.1 Detailed Description

Update Complete Status table.

9.23.2 OMA DM Update Complete Status (Update Complete Status - Meaning - Usage)

- 200 - Successful - The request has succeeded
- 250-299 - Successful(vendor specified) - successful operation with vendor specified ResultCode
- 400 - Management Client Error - Management Client error - based on User or Device behavior
- 401 - User Cancelled - User chose not to accept the operation when prompted
- 402 - Corrupted Firmware Update Package - Corrupted firmware update package did not store correctly. Detected for example, by mismatch CRCs between actual and expected
- 403 - Firmware UpdatePackage(Device Mismatch) - Wrong firmware update package delivered to device based on current device characteristics
- 404 - Failed Firmware Update Package Validation - Failure to positively validate digital signature of firmware update package
- 405 - Firmware Update Package Not acceptable - firmware update package is not acceptable
- 406 - Alternate Download Authentication Failure - authentication required but authentication failure was encountered when downloading firmware update package
- 407 - Alternate Download Request Timeout - client has encountered a timeout when downloading firmware update package
- 408 - Not Implemented - the device does not support the requested operation
- 409 - Undefined Error - indicates failure not defined by any other error code
- 410 - Firmware Update Failed - firmware update operation failed in device
- 411 - Malformed or Bad URL - the URL provided for alternate download is bad
- 412 - Alternate Download Server Unavailable - the alternate download server is unavailable or does not respond
- 450 - Client Error (OMADM General) - Vendor defined client error
- 451 - Client Error (OMADM SyncML) - Vendor defined client error
- 452 - Client Error (OMADM Auth) - Vendor defined client error
- 453 - Client Error (OMADM Protocol)- Vendor defined client error

- 454 - Client Error (OMADM Tree) - Vendor defined client error
- 455 - Client Error (OMADM DStore) - Vendor defined client error
- 456 - Client Error (OMADM Trigger) - Vendor defined client error
- 457 - Client Error (OMADM Fumo) - Vendor defined client error
- 458 - Client Error (OMADM Comms) - Vendor defined client error
- 459 - Client Error (OMADM Parse) - Vendor defined client error
- 460 - Client Error (OMADM TNDS) - Vendor defined client error
- 461 - Client Error (OMADM SCM) - Vendor defined client error
- 462 - Client Error (OMADM Impl) - Vendor defined client error
- 463-499 - Client Error (Vendor Specified) - client error encountered for operation with vendor specified result code
- 500 - Alternate Download Server Error - Alternate download server error encountered
- 501 - Download fails due to device out of memory - The download fails due to insufficient memory in the device to save the firmware update package
- 502 - Firmware update fails due to device out of memory - The update fails because there isn't sufficient memory to update the device
- 503 - Download fails due to network issues - The download fails due to network/transport level errors
- 550-599 - Alternate Download Server Error (vendor specified)- Alternate download server error encountered for operation with vendor specified result code

Copyright: © 2013 Sierra Wireless, Inc. all rights reserved

9.24 qaGobiApiTableVoiceCallEndReasons.h File Reference

Voice Service Call and supplementary services end reasons.

9.24.1 Detailed Description

Voice Service Call and supplementary services end reasons.

9.24.2 Voice Call and supplementary services end reason codes (Code - Reason)

9.24.2.1 General

- 0 - Phone is offline
- 20 - Phone is CDMA locked until a power cycle; CDMA only
- 21 - Phone has no service, this is for backward compatibility
- 22 - Call has ended abnormally; CDMA only
- 23 - Received intercept from the base station; originating only; CDMA only
- 24 - Received reorder from the base station; originating only; CDMA only
- 25 - Received release from the base station; no reason was given

- 26 - Received release from the base station; SO reject; CDMA only
- 27 - Received incoming call from the base station
- 28 - Received alert stop from the base station; incoming only; CDMA only
- 29 - Client ended the call
- 30 - Received end activation; OTASP call only; CDMA only
- 31 - MC aborted the origination/conversation; CDMA only
- 32 - Maximum access probes were transmitted; CDMA only
- 33 - Persistence test failure; FEATURE_JCDMA only; CDMA only
- 34 - R-UIM is not present
- 35 - Access attempt is already in progress
- 36 - Access failure for a reason other than the above
- 37 - Received retry order; originating only; IS 2000; CDMA only
- 38 - BYBS Concurrent service is not supported by the base station
- 39 - No response was received from the base station
- 40 - Call was rejected by the base station; CDMA only
- 41 - Concurrent services requested were not compatible; CDMA only
- 42 - Access is blocked by the base station; CDMA only
- 43 - Corresponds to CM_CALL_ORIG_ ERR_ALREADY_IN_TC
- 44 - Call is ended because an emergency call is flashed over this call; CDMA only
- 45 - Used if CM is ending a GPS call in preference of a user call
- 46 - Used if CM is ending an SMS call in preference of a user call
- 47 - Used if CM is ending a data call in preference of an emergency call
- 48 - Call was rejected because of a redirection or handoff
- 49 - Access is blocked by the base station for all mobiles; KDDI-specific; CDMA only
- 50 - To support OTASP SPC Error indication
- 51 - Maximum access probes for an IS-707B call; CDMA only
- 52 - Base station reject order
- 53 - Base station retry order
- 54 - Timer T42 is expired
- 55 - Timer T40 is expired
- 56 - Service initialization failure - Traffic Channel Initialization
- 57 - Timer T50m is expired - Traffic Channel Initialization
- 58 - Timer T51m is expired - Traffic Channel Initialization
- 59 - Acknowledgement timeout due to 12 retransmissions
- 60 - Bad forward link or timer T5M is expired
- 61 - Transceiver Resource Manager request failed

- 62 - Timer T41 is expired
- 100 - WCDMA/GSM/TDS only; call end LL cause, Received a reason for ending the call from the lower layer
- 101 - WCDMA/GSM only; Call origination request failed
- 102 - WCDMA/GSM only; client rejected an incoming call
- 103 - WCDMA/GSM only; client rejected a setup indication
- 104 - WCDMA/GSM only; network ended the call
- 105 - WCDMA/GSM only
- 106 - GWM/WCDMA only; phone has no service
- 107 - 1X only; phone has no service
- 108 - Full service is unavailable
- 109 - Indicates resources are not available to handle a new MO/MT PS call

9.24.2.2 service Errors

- 110 - Unknown subscriber
- 111 - Illegal subscriber
- 112 - Bearer service not provisioned
- 113 - Tele service not provisioned
- 114 - Illegal equipment
- 115 - Call barred
- 116 - Illegal ss operation
- 117 - Ss error status
- 118 - Ss not available
- 119 - Ss subscription violation
- 120 - Ss incompatibility
- 121 - Facility not supported
- 122 - Absent subscriber
- 123 - Short term denial
- 124 - Long term denial
- 125 - System failure
- 126 - Data missing
- 127 - Unexpected data value
- 128 - Pwd registration failure
- 129 - Negative pwd check
- 130 - Num of pwd attempts violation
- 131 - Position method failure
- 132 - Unknown alphabet

- 133 - Ussd busy
- 134 - Rejected by user
- 135 - Rejected by network
- 136 - Deflection to served subscriber
- 137 - Special service code
- 138 - Invalid deflected to number
- 139 - Mpty participants exceeded
- 140 - Resources not available

9.24.2.3 control cause values

- 141 - Unassigned number
- 142 - No route to destination
- 143 - Channel unacceptable
- 144 - Operator determined barring
- 145 - Normal call clearing
- 146 - User busy sEE [s3, aNNEX h]
- 147 - No user responding sEE [s3, aNNEX h]
- 148 - User alerting no answer
- 149 - Call rejected sEE [s3, aNNEX h]
- 150 - Number changed sEE [s3, aNNEX h]
- 151 - Preemption sEE [s3, aNNEX h]
- 152 - Destination out of order
- 153 - Invalid number format
- 154 - Facility rejected
- 155 - Resp to status enquiry
- 156 - Normal unspecified
- 157 - No circuit or channel available
- 158 - Network out of order
- 159 - Temporary failure
- 160 - Switching equipment congestion
- 161 - Access information discarded
- 162 - Requested circuit or channel not available
- 163 - Resources unavailable or unspecified
- 164 - Qos unavailable
- 165 - Requested facility not subscribed
- 166 - Incoming calls barred within cug

- 167 - Bearer capability not auth
- 168 - Bearer capability unavailable
- 169 - Service option not available
- 170 - Acm limit exceeded
- 171 - Bearer service not implemented
- 172 - Requested facility not implemented
- 173 - Only digital information bearer available
- 174 - Service or option not implemented
- 175 - Invalid transaction identifier
- 176 - USER NOT MEMBER OF CUG
- 177 - Incompatible destination
- 178 - Invalid transit nw selection
- 179 - Semantically incorrect message
- 180 - Invalid mandatory information
- 181 - Message type non implemented
- 182 - Message type not compatible with protocol statE
- 183 - Information element non existent
- 184 - Conditonal ie error
- 185 - Message not compatible with protocol state
- 186 - Recovery on timer expired
- 187 - Protocol error unspecified
- 188 - Interworking unspecified
- 189 - Outgoing calls barred within cug
- 190 - No cug selection
- 191 - Unknown cug index
- 192 - Cug index incompatible
- 193 - Cug call failure unspecified
- 194 - Clir not subscribed
- 195 - Ccbs possible sEE
- 196 - Ccbs not possible

9.24.2.4 reject causes

- 197 - Imsi unknown in hlr
- 198 - Illegal ms
- 199 - Imsi unknown in vlr
- 200 - Imei not accepted
- 201 - Illegal me sEE
- 202 - Plmn not allowed
- 203 - Location area not allowed
- 204 - Roaming not allowed in this location area
- 205 - No suitable cells in location area
- 206 - Network failure sEE
- 207 - mac failure sEE
- 208 - Synch failure
- 209 - Network congestion
- 210 - GSM authentication unacceptable
- 211 - Service not subscribed
- 212 - Service temporarily out of order
- 213 - Call cannot be identified
- 214 - Incorrect semantics in message
- 215 - Mandatory information invalid
- 216 - Call failed due to other access stratum failures
- 217 - SIM is invalid
- 218 - Invalid call state
- 219 - Access class is blocked
- 220 - No resources are in the protocol stack to allow the call
- 221 - Invalid user data was received

9.24.2.5 reject causes

- 222 - Timer T3230 is expired
- 223 - No cell is available
- 224 - Abort message was received
- 225 - Radio link was lost due to other lower layer causes

9.24.2.6 reject causes

- 226 - Timer T303 is expired
- 227 - CNM MM release is pending

9.24.2.7 stratum reject causes

- 228 - Access stratum RR release indication
- 229 - Access stratum random access failure
- 230 - RRC_REL_IND Access stratum RRC release indication
- 231 - Access stratum close session indication
- 232 - Access stratum open session failure
- 233 - Access stratum low level failure
- 234 - Access stratum low level failure redial is not allowed
- 235 - Access stratum low level immediate retry
- 236 - Access stratum abort radio is unavailable

9.24.2.8 reject causes

- 237 - Service option is not supported

9.24.2.9 IP end reasons

- 300 - Received SIP 400 bad request;waiting for INVITE response
- 301 - Received SIP 400 bad request;waiting for INVITE response
- 302 - Received SIP 404 not found; call failed; called party does not exist
- 303 - Received SIP 415 unsupported media type; call failed; called party does not support media
- 304 - Received SIP 480 temporarily unavailable; call failed; called party is not in the LTE area
- 305 - No network response; call failed
- 306 - No network response; unable to put call on hold
- 307 - Moved to eHRPD; call failed or dropped; not in the LTE area
- 308 - Upgrade/downgrade rejected (200 OK with the current call SDP)
- 309 - Received 403 call forbidden; waiting for INVITE response
- 310 - Generic timeout; did not receive a response from the server or other end
- 311 - Reported on the MO side for generic internal software errors; user can try again if the call still exists
- 312 - Reported on the MT side if the upgrade timer has been cancelled or cannot complete the request for some reason after notifying the user of a re-invite request
- 313 - Call origination is rejected due to a Service-Specific Access Control (SSAC) barring
- 314 - Phone was put in thermal emergency
- 315 - 1XCSFB call ended because of a soft failure
- 316 - 1XCSFB call ended because of a hard failure

Copyright: © 2012 Sierra Wireless, Inc. all rights reserved

9.25 qmerrno.h File Reference

Enumerations

- enum eQCWWANError {
  eQCWWAN_ERR_ENUM_BEGIN = -1,
  eQCWWAN_ERR_NONE,
  eQCWWAN_ERR_GENERAL,
  eQCWWAN_ERR_INTERNAL,
  eQCWWAN_ERR_MEMORY,
  eQCWWAN_ERR_INVALID_ARG,
  eQCWWAN_ERR_BUFFER_SZ,
  eQCWWAN_ERR_NO_DEVICE,
  eQCWWAN_ERR_INVALID_DEVID,
  eQCWWAN_ERR_NO_CONNECTION,
  eQCWWAN_ERR_QMI_IFACE,
  eQCWWAN_ERR_QMI_CONNECT,
  eQCWWAN_ERR_QMI_REQ_SCH,
  eQCWWAN_ERR_QMI_REQ,
  eQCWWAN_ERR_QMI_RSP,
  eQCWWAN_ERR_QMI_REQ_TO,
  eQCWWAN_ERR_QMI_RSP_TO,
  eQCWWAN_ERR_MALFORMED_QMI_RSP,
  eQCWWAN_ERR_INVALID_QMI_RSP,
  eQCWWAN_ERR_INVALID_FILE,
  eQCWWAN_ERR_FILE_OPEN,
  eQCWWAN_ERR_FILE_COPY,
  eQCWWAN_ERR_OFFLINE = 27,
  eQCWWAN_ERR_RESET,
  eQCWWAN_ERR_NO_SIGNAL,
  eQCWWAN_ERR_MULTIPLE_DEVICES,
  eQCWWAN_ERR_DRIVER,
  eQCWWAN_ERR_NO_CANCELABLE_OP,
  eQCWWAN_ERR_CANCEL_OP,
  eQCWWAN_ERR_API_MUTEX_TIMEOUT,
  eQCWWAN_ERR_PDU_GENERATION,
  eQCWWAN_ERR_INVALID_XID,
  eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED,
  eQCWWAN_ERR_ENUM_END,
  eQCWWAN_ERR_QMI_OFFSET = 1000,
  eQCWWAN_ERR_QMI_MALFORMED_MSG = 1001,
  eQCWWAN_ERR_QMI_NO_MEMORY,
  eQCWWAN_ERR_QMI_INTERNAL,
  eQCWWAN_ERR_QMI_ABORTED,
  eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED,
  eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION,
  eQCWWAN_ERR_QMI_INVALID_CLIENT_ID,
  eQCWWAN_ERR_QMI_NO_THRESHOLDS,
  eQCWWAN_ERR_QMI_INVALID_HANDLE,
  eQCWWAN_ERR_QMI_INVALID_PROFILE,
  eQCWWAN_ERR_QMI_INVALID_PINID,
  eQCWWAN_ERR_QMI_INCORRECT_PIN,
  eQCWWAN_ERR_QMI_NO_NETWORK_FOUND,
  eQCWWAN_ERR_QMI_CALL_FAILED,
  eQCWWAN_ERR_QMI_OUT_OF_CALL,
  eQCWWAN_ERR_QMI_NOT_PROVISIONED,
  eQCWWAN_ERR_QMI_MISSING_ARG,
  eQCWWAN_ERR_QMI_ARG_TOO_LONG = 1019,
  eQCWWAN_ERR_QMI_INVALID_TX_ID = 1022,
  eQCWWAN_ERR_QMI_DEVICE_IN_USE,
  eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED,
  eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED,
  eQCWWAN_ERR_QMI_NO_EFFECT,
  eQCWWAN_ERR_QMI_NO_FREE_PROFILE,
  eQCWWAN_ERR_QMI_INVALID_PDP_TYPE,
  eQCWWAN_ERR_QMI_INVALID_TECH_PREF,
  eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE,

eQCWWAN_ERR_QMI_WIDTH = 0xFFFF }

- enum qm_wds_ds_profile_extended_err_codes {
  eWDS_ERR_PROFILE_REG_RESULT_FAIL = 1,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID,
  eWDS_ERR_PROFILE_REG_RESULT_LIST_END,
  eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID,
  eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY,
  eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY = 1001,
  eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR,
  eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED,
  eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET,
  eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET,
  eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES,
  eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE = 1101,
  eWDS_ERR_PROFILE_REG_END }

9.25.1 Enumeration Type Documentation

9.25.1.1 enum eQCWWANError

QMI Error Code Enumeration

Enumerator

eQCWWAN_ERR_ENUM_BEGIN

eQCWWAN_ERR_NONE 00 - Success

eQCWWAN_ERR_GENERAL 01 - General error

eQCWWAN_ERR_INTERNAL 02 - Internal error

eQCWWAN_ERR_MEMORY 03 - Memory error

eQCWWAN_ERR_INVALID_ARG 04 - Invalid argument

eQCWWAN_ERR_BUFFER_SZ 05 - Buffer too small

eQCWWAN_ERR_NO_DEVICE 06 - Unable to detect WWAN device

eQCWWAN_ERR_INVALID_DEVID 07 - Invalid WWAN device ID

eQCWWAN_ERR_NO_CONNECTION 08 - No connection to WWAN device

eQCWWAN_ERR_QMI_IFACE 09 - Unable to obtain QMI interace

eQCWWAN_ERR_QMI_CONNECT 10 - Unable to connect to QMI interface

eQCWWAN_ERR_QMI_REQ_SCH 11 - Unable to schedule QMI request

eQCWWAN_ERR_QMI_REQ 12 - Error sending QMI request

eQCWWAN_ERR_QMI_RSP 13 - Error receiving QMI response

eQCWWAN_ERR_QMI_REQ_TO 14 - Timeout while sending QMI request

eQCWWAN_ERR_QMI_RSP_TO 15 - Timeout while receiving QMI response

eQCWWAN_ERR_MALFORMED_QMI_RSP 16 - Malformed QMI response received

eQCWWAN_ERR_INVALID_QMI_RSP 17 - Invalid QMI response received

eQCWWAN_ERR_INVALID_FILE 18 - Invalid file path

eQCWWAN_ERR_FILE_OPEN 19 - Unable to open file

eQCWWAN_ERR_FILE_COPY 20 - Unable to copy file

eQCWWAN_ERR_OFFLINE 27 - Unable to set WWAN device offline

eQCWWAN_ERR_RESET 28 - Unable to reset WWAN device

eQCWWAN_ERR_NO_SIGNAL 29 - No available signal

eQCWWAN_ERR_MULTIPLE_DEVICES 30 - Multiple WWAN devices detected

eQCWWAN_ERR_DRIVER 31 - Error interfacing to driver

eQCWWAN_ERR_NO_CANCELABLE_OP 32 - No cancelable operation is pending

eQCWWAN_ERR_CANCEL_OP 33- Error canceling outstanding operation

eQCWWAN_ERR_API_MUTEX_TIMEOUT 34- api mutex lock timeout

eQCWWAN_ERR_PDU_GENERATION 35- PDU generation error

eQCWWAN_ERR_INVALID_XID 36- Invalid transaction id

eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED 37- Unsupported multiple SMS

eQCWWAN_ERR_ENUM_END End of SLQS SDK specific error codes

eQCWWAN_ERR_QMI_OFFSET 1000 - This is not an error code but the offset from which mapped QMI error codes start from

eQCWWAN_ERR_QMI_MALFORMED_MSG 1001 - Malformed or Corrupted QMI msg

eQCWWAN_ERR_QMI_NO_MEMORY 1002 - Device could not allocate memory for QMI Resp

eQCWWAN_ERR_QMI_INTERNAL 1003 - Unexpected error occurred during processing

eQCWWAN_ERR_QMI_ABORTED 1004 - Processing aborted

eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED 1005 - QMI client IDs have been exhausted

eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION 1006 - Unable to abort QMI transaction

eQCWWAN_ERR_QMI_INVALID_CLIENT_ID 1007 - Invalid QMI client ID

eQCWWAN_ERR_QMI_NO_THRESHOLDS 1008 - No thresholds were provided

eQCWWAN_ERR_QMI_INVALID_HANDLE 1009 - Invalid Handle provided in the QMI request

eQCWWAN_ERR_QMI_INVALID_PROFILE 1010 - Profile specified is invalid

eQCWWAN_ERR_QMI_INVALID_PINID 1011 - Invalid PIN ID specified

eQCWWAN_ERR_QMI_INCORRECT_PIN 1012 - Incorrect PIN ID specified

eQCWWAN_ERR_QMI_NO_NETWORK_FOUND 1013 - No network found

eQCWWAN_ERR_QMI_CALL_FAILED 1014 - Call failed

eQCWWAN_ERR_QMI_OUT_OF_CALL 1015 - Device is not in a call

eQCWWAN_ERR_QMI_NOT_PROVISIONED 1016 - Requested information element not provisioned on device

eQCWWAN_ERR_QMI_MISSING_ARG 1017 - Mandatory QMI TLV not provided

eQCWWAN_ERR_QMI_ARG_TOO_LONG 1019 - Arg passed in QMI TLV larger than available storage in device

eQCWWAN_ERR_QMI_INVALID_TX_ID 1022 - Invalid TX ID specified

eQCWWAN_ERR_QMI_DEVICE_IN_USE 1023 - Device currently in a call

eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED 1024 - The selected operation is not supported by the network

eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED 1025 - The selected operation is not supported by the device

eQCWWAN_ERR_QMI_NO_EFFECT 1026 - Requested operation would have no effect

eQCWWAN_ERR_QMI_NO_FREE_PROFILE 1027 - No space for a profile is available

eQCWWAN_ERR_QMI_INVALID_PDP_TYPE 1028 - Invalid PDP type specified

eQCWWAN_ERR_QMI_INVALID_TECH_PREF 1029 - Invalid technology preference specified

eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE 1030 - Invalid profile type specified

eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE 1031 - Invalid service type specified

eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION 1032 - Invalid register action specified

eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION 1033 - Invalid PS attach/detach action specified

eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED 1034 - Authentication of supplied information element failed

eQCWWAN_ERR_QMI_PIN_BLOCKED 1035 - PIN is blocked; an unblock operation needs to be issued

eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED 1036 - PIN is permanently blocked; the UIM is unusable

eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED 1037 - UIM initialization has not completed

eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE 1038 - Max QOS requests are used

eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER 1039 - The Flow filter is incorrect

eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE 1040 - Network unaware of the QOS requested

eQCWWAN_ERR_QMI_INVALID_ID 1041 - Invalid QOS ID

eQCWWAN_ERR_QMI_INVALID_QOS_ID 1041 - Invalid QOS ID

eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED 1042 - The request number is not supported

eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND 1043 - Unable to find the interface

eQCWWAN_ERR_QMI_FLOW_SUSPENDED 1044- Flow suspended

eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT 1045 - Data format is invalid

eQCWWAN_ERR_QMI_GENERAL 1046 - General error

eQCWWAN_ERR_QMI_UNKNOWN 1047 - Unknown error

eQCWWAN_ERR_QMI_INVALID_ARG 1048 - A specified argument is invalid

eQCWWAN_ERR_QMI_INVALID_INDEX 1049 - A specified index is invalid

eQCWWAN_ERR_QMI_NO_ENTRY 1050 - No information element exists at specified memory designation

eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL 1051 - The memory storage specified in the request is full

eQCWWAN_ERR_QMI_DEVICE_NOT_READY 1052 - Device not in a ready state

eQCWWAN_ERR_QMI_NETWORK_NOT_READY 1053 - Network not in a ready state

eQCWWAN_ERR_QMI_CAUSE_CODE 1054 - Error provided in SMS cause code

eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT 1055 - The message could not be sent

eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE 1056 - The message could not be delivered

eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID 1057 - The message ID specified for the message is invalid

eQCWWAN_ERR_QMI_ENCODING 1058 - The message is not encoded properly

eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK 1059 - Maximum number of authentication failures has been reached

eQCWWAN_ERR_QMI_INVALID_TRANSITION 1060 - Operating mode transition from the current mode is invalid

eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE 1061 - The intercace is not muticast

eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE 1062 - Maximum requests in use

eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE 1063 - Invalid muticast handle

eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF 1064 - Invalid IP family preference

eQCWWAN_ERR_QMI_SESSION_INACTIVE 1065 - No tracking session has been started

eQCWWAN_ERR_QMI_SESSION_INVALID 1066 - Current session does not allow this operation

eQCWWAN_ERR_QMI_SESSION_OWNERSHIP 1067 - Current tracking session not started by this QMI control point

eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES 1068 - Device GPS service resources insufficient for request

eQCWWAN_ERR_QMI_DISABLED 1069 - Device GPS service disabled

eQCWWAN_ERR_QMI_INVALID_OPERATION 1070 - Invalid operation specified

eQCWWAN_ERR_QMI_INVALID_QMI_CMD 1071 - Invalid/unknown QMI command specified

eQCWWAN_ERR_QMI_TPDU_TYPE 1072 - Message contains TPDU type that cannot be read as raw message

eQCWWAN_ERR_QMI_SMSC_ADDR 1073 - The SMSC address specified is invalid

eQCWWAN_ERR_QMI_INFO_UNAVAILABLE 1074 - Information element is unavailable at this point

eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG 1075 - Segment size too large

eQCWWAN_ERR_QMI_SEGMENT_ORDER 1076 - Segment order is incorrect

eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED 1077 - Bundling not supported

eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE 1078 - The operation failed partially

eQCWWAN_ERR_QMI_POLICY_MISMATCH 1079 - Policy mismatch

eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND 1080 - SIM file not found

eQCWWAN_ERR_QMI_EXTENDED_INTERNAL 1081 - Extended internal error

eQCWWAN_ERR_QMI_ACCESS_DENIED 1082 - Access to a required entity is not available

eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED 1083 - Selected operating mode is invalid with current hardware setting

eQCWWAN_ERR_QMI_ACK_NOT_SENT 1084 - ACK not sent

eQCWWAN_ERR_QMI_INJECT_TIMEOUT 1084 - Inject a timeout for the request

eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE 1090 - Incompatible state

eQCWWAN_ERR_QMI_FDN_RESTRICT 1091 - FDN Restrict

eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE 1092 - SUPS failure cause

eQCWWAN_ERR_QMI_NO_RADIO 1093 - No Radio

eQCWWAN_ERR_QMI_NOT_SUPPORTED 1094 - Not Supported

eQCWWAN_ERR_QMI_NO_SUBSCRIPTION 1095 - No Subscription

eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED 1096 - Card call control failed

eQCWWAN_ERR_QMI_NETWORK_ABORTED 1097 - Network Aborted

eQCWWAN_ERR_QMI_MSG_BLOCKED 1098 - Open Error

eQCWWAN_ERR_QMI_MAX Error - End of QMI specific defines

eQCWWAN_ERR_SWICM_START **Vendor defines - Connection Manager error codes**

eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED 0xE001 - The API is yet to be implemented

eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED 0xE002 - The service is not supported

eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED 0xE003 - The client is not supported

eQCWWAN_ERR_SWICM_TIMEOUT 0xE004 - API Timeout

eQCWWAN_ERR_SWICM_SOCKET_IN_USE 0xE005 - The communication socket is in use

eQCWWAN_ERR_SWICM_AM_VERS_ERROR 0xE006 - SLQS API and SDK version mismatch

eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS 0xE007 - Failed to kill SDK process

eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS 0xE008 - Call in progress

eQCWWAN_ERR_SWICM_V4DWN_V6DWN 0xE009 - IPV4 and IPV6 is down

eQCWWAN_ERR_SWICM_V4DWN_V6UP 0xE00A - IPV4 is down and IPV6 is up

eQCWWAN_ERR_SWICM_V4UP_V6DWN 0xE00B - IPV4 is up and IPV6 is down

eQCWWAN_ERR_SWICM_V4UP_V6UP 0xE00C - IPV4 and IPV6 is up

eQCWWAN_ERR_SWICM_INVALID_SESSION_ID 0xE00D - Invalid V4 Session ID

eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID 0xE00E - Invalid V4 Session ID

eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID 0xE00F - Invalid V6 Session ID

eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS 0xE010 - No available Session Manager slots for additional data sessions

eQCWWAN_ERR_SWICM_END 0xE011 - End of connection manager specific codes

eQCWWAN_ERR_SWISMS_START **Vendor defines - SMS Error codes**

eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG 0xE101 - SMS message length is long

eQCWWAN_ERR_SWISMS_MSG_CORRUPTED 0xE102 - The SMS message is corrupted (encoding wrong)

eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED 0xE103 - The SMS number is corrupted (incorrect number)

eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND 0xE104 - The SMS bearer data is not available

eQCWWAN_ERR_SWISM_END

eQCWWAN_ERR_SWIIM_START **Vendor defines - Image Management error codes**

eQCWWAN_ERR_SWIIM_INVALID_PATH 0xE801 - Invalid directory path

eQCWWAN_ERR_SWIIM_OPENING_DIR 0xE802 - Unable to open the directory

eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND 0xE803 - No Firmware image present in the path

eQCWWAN_ERR_SWIIM_OPENING_FILE 0xE804 - Unable to open the file

eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE 0xE805 - Firmware image is corrupted

eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED 0xE806 - No Firmware image download needed

eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL 0xE807 - Firmware update failed

eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH 0xE808 - Update success but pri/fw preference mismatch

eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS 0xE809 - Update successful

eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE 0xE80A - Enter Download Mode

eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE 0xE80B - File transfer to modem complete

eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT 0xE80C - Wait for modem to reboot

eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE 0xE80D - Invalid Crash State for Firmware Download

eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE 0xE80E - Same as current active image

eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX 0xE80F - invalid slot index

eQCWWAN_ERR_SWIIM_END

eQCWWAN_ERR_SWIDCS_START **Vendor defines - Device Connectivity error codes**

eQCWWAN_ERR_SWIDCS_IOCTL_ERR 0xE901 - IO Control error

eQCWWAN_ERR_SWIDCS_FILEIO_ERR 0xE902 - file open/read/write error

eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND 0xE903 - The device is not found

eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED 0xE904 - Application is disconnected from SDK

eQCWWAN_ERR_SWIDCS_END

eQCWWAN_ERR_QMI_CAT_START **QMI errors related to CAT**

eQCWWAN_ERR_QMI_EVENT_REG_FAILED 62441 - CAT event registration failed

eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP 62442 - Invalid terminal response

eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD 62443 - Invalid envelope command

eQCWWAN_ERR_QMI_CARD_BUSY_RSP 62444 - Card busy response for envelope command

eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE 62445 - Envelope command failure

eQCWWAN_ERR_QMI_CAT_END

eQCWWAN_ERR_NULL_TLV

eQCWWAN_ERR_QMI_WIDTH 0xFFFF - Not an error, represent the end of QMI errors

9.25.1.2 enum qm_wds_ds_profile_extended_err_codes

WDS DS profile extented error codes

Enumerator

eWDS_ERR_PROFILE_REG_RESULT_FAIL 1 - General Failure

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL 2 - The request contains an invalid profile handle

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP 3 - An invalid operation was requested.

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE 4 - The request contains an invalid technology type

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM 5 - The request contains an invalid profile number

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT 6 - The request contains an invalid profile identifier

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL 7 - The request contains an invalid argument other than profile number and profile identifier received.

eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED 8 - Profile registry has not been initialized yet

eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID 9 - The request contains a parameter with invalid length.

eWDS_ERR_PROFILE_REG_RESULT_LIST_END 10 - End of the profile list was reached while searching for the requested profile.

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID 11 - The request contains an invalid subscrition identifier.

eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY 12 - The request contains an invalid profile family.

eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY 1001 - The request contains an invalid 3GPP profile family.

eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR 1002 - An error was encountered while accessing the 3GPP profiles.

eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED 1003 - The given 3GPP profile doesn't have a valid context.

eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET 1004 - The given 3GPP profile is marked invalid.

eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET 1005 - The given 3GPP profile is marked read-only.

eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES 1006 - Creation of a new 3GPP profile failed because the limit of 16 profiles has already been reached.

eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE 1101 - An invalid profile identifier was received as part of the 3GPP2 profile modification request.

eWDS_ERR_PROFILE_REG_END

9.26 qos.h File Reference

Data Structures

- struct unpack_qos_SLQSQosGetNetworkStatus_t
- struct pack_qos_SLQSQosSwiReadApnExtraParams_t
- struct unpack_qos_SLQSQosSwiReadApnExtraParams_t

- struct pack_qos_SLQSQosSwiReadDataStats_t
- struct unpack_QosFlowStat_t
- struct unpack_qos_SLQSQosSwiReadDataStats_t
- struct unpack_qos_SLQSSetQosNWStatusCallback_ind_t
- struct unpack_qos_SLQSSetQosStatusCallback_ind_t
- struct unpack_qos_SLQSSetQosPriEventCallback_ind_t
- struct pack_qos_SLQSSetQosEventCallback_t
- struct unpack_qos_QosFlowInfoState_t
- struct unpack_qos_dataRate_t
- struct unpack_qos_tokenBucket_t
- struct unpack_qos_pktErrRate_t
- struct unpack_qos_swiQosFlow_t
- struct unpack_qos_IPv4Addr_t
- struct unpack_qos_Tos_t
- struct unpack_qos_IPv6Addr_t
- struct unpack_qos_IPv6TrafCls_t
- struct unpack_qos_Port_t
- struct unpack_qos_swiQosFilter_t
- struct unpack_qos_QosFlowInfo_t
- struct unpack_qos_SLQSSetQosEventCallback_ind_t

Macros

- #define LITEQMI_MAX_QOS_FLOW_PER_APN_STATS 10
- #define LITEQMI_MAX_QOS_FILTERS 25
- #define LITEQMI_MAX_QOS_FLOWS 8

Functions

- int pack_qos_SLQSQosGetNetworkStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_qos_SLQSQosGetNetworkStatus (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosGetNetworkStatus_t *pOutput)
- int pack_qos_SLQSQosSwiReadApnExtraParams (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSQosSwiReadApnExtraParams_t reqParam)
- int unpack_qos_SLQSQosSwiReadApnExtraParams (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosSwiReadApnExtraParams_t *pOutput)
- int pack_qos_SLQSQosSwiReadDataStats (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSQosSwiReadDataStats_t reqParam)
- int unpack_qos_SLQSQosSwiReadDataStats (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSQosSwiReadDataStats_t *pOutput)
- int unpack_qos_SLQSSetQosNWStatusCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosNWStatusCallback_ind_t *pOutput)
- int unpack_qos_SLQSSetQosStatusCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosStatusCallback_ind_t *pOutput)
- int unpack_qos_SLQSSetQosPriEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosPriEventCallback_ind_t *pOutput)
- int pack_qos_SLQSSetQosEventCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_qos_SLQSSetQosEventCallback_t reqParam)
- int unpack_qos_SLQSSetQosEventCallback (uint8_t *pResp, uint16_t respLen)
- int unpack_qos_SLQSSetQosEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_qos_SLQSSetQosEventCallback_ind_t *pOutput)

9.26.1 Macro Definition Documentation

9.26.1.1 #define LITEQMI_MAX_QOS_FILTERS 25

9.26.1.2 #define LITEQMI_MAX_QOS_FLOW_PER_APN_STATS 10

9.26.1.3 #define LITEQMI_MAX_QOS_FLOWS 8

9.26.2 Function Documentation

9.26.2.1 int pack_qos_SLQSQosGetNetworkStatus (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen*)

Function to pack command to retrieve QoS status of the network. This maps to SLQSQosGetNetworkStatus

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
- Technology Supported: CDMA
- PDN Specific: No

9.26.2.2 int pack_qos_SLQSQosSwiReadApnExtraParams (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* pack_qos_SLQSQosSwiReadApnExtraParams_t *reqParam*)

Function to pack QMI command to query extra APN parameters This maps to SLQSQosSwiReadApnExtraParams

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes

in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_qos_SLQSQosSwiReadApnExtraParams_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
  - PDN Specific: Yes

9.26.2.3 int pack_qos_SLQSQosSwiReadDataStats (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_qos_SLQSQosSwiReadDataStats_t *reqParam*)

Function to pack QMI command to query APN data statistics This maps to SLQSQosSwiReadDataStats

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_qos_SLQSQosSwiReadDataStats_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
  - PDN Specific: Yes

9.26.2.4 int pack_qos_SLQSSetQosEventCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_qos_SLQSSetQosEventCallback_t *reqParam*)

Function to pack QMI command to enable QoS event indications This maps to SLQSSetQosEventCallback

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_qos_SLQSSetQosEventCallback_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Timeout: 2 seconds
  - PDN Specific: Yes

9.26.2.5 int unpack_qos_SLQSQosGetNetworkStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSQosGetNetworkStatus_t * *pOutput*)

Function to unpack the response to get NW QoS status command This maps to SLQSQosGetNetworkStatus

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSQosGetNetworkStatus_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.26.2.6 int unpack_qos_SLQSQosSwiReadApnExtraParams (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSQosSwiReadApnExtraParams_t * *pOutput*)

Function to unpack the response to get NW QoS status command This maps to SLQSQosSwiReadApnExtraParams

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSQosSwiReadApnExtraParams_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.26.2.7 int unpack_qos_SLQSQosSwiReadDataStats (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSQosSwiReadDataStats_t * *pOutput*)

Function to unpack APN data statistics response This maps to SLQSQosSwiReadDataStats

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSQosSwiReadDataStats_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.26.2.8 int unpack_qos_SLQSSetQosEventCallback (uint8_t * *pResp*, uint16_t *respLen*)

Function to unpack enable QoS event indications command's response This maps to SLQSSetQosEventCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.26.2.9 int unpack_qos_SLQSSetQosEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosEventCallback_ind_t * *pOutput*)

Function to unpack QoS event indications This maps to SLQSSetQosEventCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosEventCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_NETWORK_STATUS_IND indication to identify this event from QOS service read function

9.26.2.10 int unpack_qos_SLQSSetQosNWStatusCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosNWStatusCallback_ind_t * *pOutput*)

Function to unpack QoS NW status indication. This maps to SLQSSetQosNWStatusCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosNWStatusCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Technology Supported: CDMA
- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_NETWORK_STATUS_IND indication to identify this event from QOS service read function

9.26.2.11 int unpack_qos_SLQSSetQosPriEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosPriEventCallback_ind_t * *pOutput*)

Function to unpack QoS primary flow events. This maps to SLQSSetQosPriEventCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosPriEventCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_PRIMARY_QOS_EVENT_IND indication to identify this event from QOS service read function
- This is only generated when the primary flow is modified by the host

9.26.2.12 int unpack_qos_SLQSSetQosStatusCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_qos_SLQSSetQosStatusCallback_ind_t * *pOutput*)

Function to unpack QoS status indications. This maps to SLQSSetQosStatusCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_qos_SLQSSetQosStatusCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- This is a broadcast notification that is sent to the registered QoS service
- No explicit function to register for this indication is needed
- Please use eQMI_QOS_FLOW_STATUS_IND indication to identify this event from QOS service read function

9.27 rms.h File Reference

Data Structures

- struct unpack_rms_GetSMSWake_t
- struct pack_rms_SetSMSWake_t

Functions

- int pack_rms_GetSMSWake (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_rms_GetSMSWake (uint8_t *pResp, uint16_t respLen, unpack_rms_GetSMSWake_t *pOutput)
- int pack_rms_SetSMSWake (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_rms_SetSMSWake_t *reqArg)
- int unpack_rms_SetSMSWake (uint8_t *pResp, uint16_t respLen)

9.27.1 Function Documentation

9.27.1.1 int pack_rms_GetSMSWake (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen*)

Get SMS Wake pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.1.2 int pack_rms_SetSMSWake (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* pack_rms_SetSMSWake_t * *reqArg*)

Set SMS Wake pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.1.3 int unpack_rms_GetSMSWake (uint8_t * *pResp,* uint16_t *respLen,* unpack_rms_GetSMSWake_t * *pOutput*)

Get SMS Wake unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.1.4 int unpack_rms_SetSMSWake (uint8_t * *pResp,* uint16_t *respLen*)

Set SMS Wake unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.28 sar.h File Reference

Data Structures

- struct unpack_sar_SLQSGetRfSarState_t
- struct pack_sar_SLQSSetRfSarState_t

Functions

- int pack_sar_SLQSGetRfSarState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sar_SLQSGetRfSarState (uint8_t *pResp, uint16_t respLen, unpack_sar_SLQSGetRfSarState_t *pOutput)
- int pack_sar_SLQSSetRfSarState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sar_SLQSSetRfSarState_t *reqArg)
- int unpack_sar_SLQSSetRfSarState (uint8_t *pResp, uint16_t respLen)

9.28.1 Function Documentation

9.28.1.1 int pack_sar_SLQSGetRfSarState (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen*)

Gets the specified RF SAR state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.28.1.2 int pack_sar_SLQSSetRfSarState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sar_SLQSSetRfSarState_t * *reqArg*)

Sets the specified RF SAR state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.28.1.3 int unpack_sar_SLQSGetRfSarState (uint8_t * *pResp*, uint16_t *respLen*, unpack_sar_SLQSGetRfSarState_t * *pOutput*)

Gets the specified RF SAR state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.28.1.4 int unpack_sar_SLQSSetRfSarState (uint8_t * *pResp*, uint16_t *respLen*)

Sets the specified RF SAR state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29 sms.h File Reference

Data Structures

- struct pack_sms_SLQSGetSMS_t
- struct unpack_sms_SLQSGetSMS_t
- struct pack_sms_SLQSGetSMSList_t
- struct qmiSmsMessageList
- struct unpack_sms_SLQSGetSMSList_t
- struct pack_sms_SLQSModifySMSStatus_t
- struct unpack_sms_SLQSModifySMSStatus_t
- struct pack_sms_SLQSDeleteSMS_t
- struct unpack_sms_SLQSDeleteSMS_t
- struct pack_sms_SendSMS_t
- struct unpack_sms_SendSMS_t
- struct pack_sms_SetNewSMSCallback_t
- struct unpack_sms_SetNewSMSCallback_t
- struct sMSMTMessage
- struct newMTMessageTlv
- struct sMSTransferRouteMTMessage
- struct transferRouteMessageTlv
- struct sMSMessageMode
- struct messageModeTlv
- struct sMSEtwsMessage
- struct sMSEtwsMessageTlv
- struct sMSEtwsPlmn
- struct eTWSPLMNInfoTlv
- struct sMSCAddress
- struct sMSCAddressTlv
- struct sMSOnIMS
- struct sMSOnIMSTlv
- struct unpack_sms_SetNewSMSCallback_ind_t
- struct unpack_sms_SLQSWmsMemoryFullCallBack_ind_t
- struct unpack_sms_GetSMSCAddress_t
- struct pack_sms_SetSMSCAddress_t
- struct pack_sms_SaveSMS_t
- struct unpack_sms_SaveSMS_t

- struct sms_BroadcastConfig
- struct sms_qaQmi3GPPBroadcastCfgInfo
- struct sms_CDMABroadcastConfig
- struct sms_qaQmi3GPP2BroadcastCfgInfo
- struct unpack_sms_SLQSGetSmsBroadcastConfig_t
- struct pack_sms_SLQSGetSmsBroadcastConfig_t
- struct pack_sms_SLQSSetSmsBroadcastConfig_t
- struct pack_sms_SLQSSetSmsBroadcastActivation_t
- struct sms_transLayerInfo
- struct sms_getTransLayerInfo
- struct unpack_sms_SLQSGetTransLayerInfo_t
- struct sms_getTransNWRegInfo
- struct unpack_sms_SLQSGetTransNWRegInfo_t
- struct sms_getIndicationReg
- struct unpack_sms_SLQSGetIndicationRegister_t
- struct sms_setIndicationReg
- struct pack_sms_SLQSSetIndicationRegister_t
- struct sms_routeEntry
- struct sms_setRoutesReq
- struct pack_sms_SLQSSmsSetRoutes_t
- struct sms_msgProtocolResp
- struct unpack_sms_SLQSSmsGetMessageProtocol_t
- struct sms_maxStorageSizeReq
- struct sms_maxStorageSizeResp
- struct pack_sms_SLQSSmsGetMaxStorageSize_t
- struct unpack_sms_SLQSSmsGetMaxStorageSize_t
- struct sms_messageWaitingInfoContent
- struct sms_getMsgWaitingInfo
- struct unpack_sms_SLQSGetMessageWaiting_t
- struct sms_sendAsyncsmsParams
- struct pack_sms_SLQSSendAsyncSMS_t
- struct pack_sms_SLQSSetSmsStorage_t
- struct unpack_sms_SLQSSwiGetSMSStorage_t

Macros

- #define MAX_SMS_MESSAGE_SIZE 2048
- #define MAX_SMS_LIST_SIZE 255
- #define MAX_MS_TRANSFER_ROUTE_MSG 256
- #define MAX_MSE_TWS_MSG 1254
- #define MAX_MSC_ADDRESS_SIZE 256
- #define MAX_CDMA_ENC_MO_TXT_MSG_SIZE 255
- #define SMSC_TYPE_LEN 0x03
- #define SMS_CONFIG_LEN 0x05
- #define SMS_MAX_SMS_ROUTES 0x0A
- #define SMS_NUM_OF_SET 0xFF

Typedefs

- typedef struct sMSMTMessage sMSMTMessageInfo
- typedef struct sMSTransferRouteMTMessage sMSTransferRouteMTMessageInfo
- typedef struct sMSMessageMode sMSMessageModeInfo
- typedef struct sMSEtwsMessage sMSEtwsMessageInfo
- typedef struct sMSEtwsPlmn sMSEtwsPlmnInfo
- typedef struct sMSCAddress sMSCAddressInfo
- typedef struct sMSOnIMS sMSOnIMSInfo

Enumerations

- enum eqmiCbkSetStatus {
  LITEQMI_QMI_CBK_PARAM_RESET = 0,
  LITEQMI_QMI_CBK_PARAM_SET = 1,
  LITEQMI_QMI_CBK_PARAM_NOCHANGE }

Functions

- int pack_sms_SLQSGetSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSMS_t *reqParam)
- int unpack_sms_SLQSGetSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSMS_t *pOutput)
- int pack_sms_SLQSGetSMSList (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSMSList_t *reqParam)
- int unpack_sms_SLQSGetSMSList (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSMSList_t *pOutput)
- int pack_sms_SLQSModifySMSStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSModifySMSStatus_t *reqParam)
- int unpack_sms_SLQSModifySMSStatus (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSModifySMSStatus_t *pOutput)
- int pack_sms_SLQSDeleteSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSDeleteSMS_t *reqParam)
- int unpack_sms_SLQSDeleteSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSDeleteSMS_t *pOutput)
- int pack_sms_SendSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SendSMS_t *reqParam)
- int unpack_sms_SendSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SendSMS_t *pOutput)
- int pack_sms_SetNewSMSCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SetNewSMSCallback_t reqParam)
- int unpack_sms_SetNewSMSCallback (uint8_t *pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_t *Output)
- int unpack_sms_SetNewSMSCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SetNewSMSCallback_ind_t *pOutput)
- int unpack_sms_SLQSWmsMemoryFullCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSWmsMemoryFullCallBack_ind_t *pOutput)
- int pack_sms_GetSMSCAddress (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_GetSMSCAddress (uint8_t *pResp, uint16_t respLen, unpack_sms_GetSMSCAddress_t *pOutput)
- int pack_sms_SetSMSCAddress (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SetSMSCAddress_t *reqParam)
- int unpack_sms_SetSMSCAddress (uint8_t *pResp, uint16_t respLen)
- int pack_sms_SaveSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SaveSMS_t *reqParam)
- int unpack_sms_SaveSMS (uint8_t *pResp, uint16_t respLen, unpack_sms_SaveSMS_t *pOutput)
- int pack_sms_SLQSGetSmsBroadcastConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSGetSmsBroadcastConfig_t *reqParam)
- int unpack_sms_SLQSGetSmsBroadcastConfig (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetSmsBroadcastConfig_t *pOutput)
- int pack_sms_SLQSSetSmsBroadcastConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetSmsBroadcastConfig_t *reqParam)
- int unpack_sms_SLQSSetSmsBroadcastConfig (uint8_t *pResp, uint16_t respLen)
- int pack_sms_SLQSSetSmsBroadcastActivation (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetSmsBroadcastActivation_t *reqParam)
- int unpack_sms_SLQSSetSmsBroadcastActivation (uint8_t *pResp, uint16_t respLen)
- int pack_sms_SLQSGetTransLayerInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)

- int unpack_sms_SLQSGetTransLayerInfo (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetTransLayerInfo_t *pOutput)
- int pack_sms_SLQSGetTransNWRegInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSGetTransNWRegInfo (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetTransNWRegInfo_t *pOutput)
- int pack_sms_SLQSGetIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSGetIndicationRegister (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetIndicationRegister_t *pOutput)
- int pack_sms_SLQSSetIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetIndicationRegister_t *reqParam)
- int unpack_sms_SLQSSetIndicationRegister (uint8_t *pResp, uint16_t respLen)
- int pack_sms_SLQSSmsSetRoutes (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSmsSetRoutes_t *reqParam)
- int unpack_sms_SLQSSmsSetRoutes (uint8_t *pResp, uint16_t respLen)
- int pack_sms_SLQSSmsGetMessageProtocol (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSSmsGetMessageProtocol (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSmsGetMessageProtocol_t *pOutput)
- int pack_sms_SLQSSmsGetMaxStorageSize (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSmsGetMaxStorageSize_t *reqParam)
- int unpack_sms_SLQSSmsGetMaxStorageSize (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSmsGetMaxStorageSize_t *pOutput)
- int pack_sms_SLQSGetMessageWaiting (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSGetMessageWaiting (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSGetMessageWaiting_t *pOutput)
- int pack_sms_SLQSSendAsyncSMS (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSendAsyncSMS_t *reqParam)
- int unpack_sms_SLQSSendAsyncSMS (uint8_t *pResp, uint16_t respLen)
- int pack_sms_SLQSSetSmsStorage (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_sms_SLQSSetSmsStorage_t *reqParam)
- int unpack_sms_SLQSSetSmsStorage (uint8_t *pResp, uint16_t respLen)
- int pack_sms_SLQSSwiGetSMSStorage (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_sms_SLQSSwiGetSMSStorage (uint8_t *pResp, uint16_t respLen, unpack_sms_SLQSSwiGetSMSStorage_t *pOutput)

9.29.1 Macro Definition Documentation

9.29.1.1 #define MAX_CDMA_ENC_MO_TXT_MSG_SIZE 255

9.29.1.2 #define MAX_MS_TRANSFER_ROUTE_MSG 256

9.29.1.3 #define MAX_MSC_ADDRESS_SIZE 256

9.29.1.4 #define MAX_MSE_TWS_MSG 1254

9.29.1.5 #define MAX_SMS_LIST_SIZE 255

9.29.1.6 #define MAX_SMS_MESSAGE_SIZE 2048

9.29.1.7 #define SMS_CONFIG_LEN 0x05

9.29.1.8 #define SMS_MAX_SMS_ROUTES 0x0A

9.29.1.9 #define SMS_NUM_OF_SET 0xFF

9.29.1.10 #define SMSC_TYPE_LEN 0x03

9.29.2 Typedef Documentation

9.29.2.1 typedef struct sMSCAddress sMSCAddressInfo

Parameters

length	• Number of sets of following element
data	• SMSC address

9.29.2.2 typedef struct sMSEtwsMessage sMSEtwsMessageInfo

Parameters

notificationType	• Message mode 0x00 - Primary 0x01 - Secondary GSM 0x02 - Secondary UMTS
length	• Number of sets of following elements
data	• Raw message data

9.29.2.3 typedef struct sMSEtwsPlmn sMSEtwsPlmnInfo

Parameters

mobileCountry-Code	• 16 bit representation of MCC value range : 0 -999
mobileNetwork-Code	• 16 bit representation of MNC value range : 0 -999

9.29.2.4 typedef struct sMSMessageMode sMSMessageModeInfo

Parameters

messageMode	Message Mode

9.29.2.5 typedef struct sMSMTMessage sMSMTMessageInfo

Parameters

storageType	memory storage 0x00-UIM 0x01-NV
messageIndex	MT Message index

9.29.2.6 typedef struct sMSOnIMS sMSOnIMSInfo

Parameters

smsOnIMS	SMS on IMS

9.29.2.7 typedef struct sMSTransferRouteMTMessage sMSTransferRouteMTMessageInfo

Parameters

ackIndicator	• Parameter to indicate if ACK must be sent by the control point 0x00 - Send ACK 0x01 - Do not send ACK
transactionID	• Transaction ID of the message
format	• Message format 0x00 - CDMA 0x02 - 0x05 - Reserved 0x06 - GW_PP 0x07 - GW_BC
length	• Length of the raw message. This length should not exceed the maximum WMS payload length of 256 bytes
data	• Raw message data

9.29.3 Enumeration Type Documentation

9.29.3.1 enum eqmiCbkSetStatus

Enumerator

LITEQMI_QMI_CBK_PARAM_RESET

LITEQMI_QMI_CBK_PARAM_SET

LITEQMI_QMI_CBK_PARAM_NOCHANGE

9.29.4 Function Documentation

9.29.4.1 int pack_sms_GetSMSCAddress (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the SMS center address pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.2 int pack_sms_SaveSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SaveSMS_t * *reqParam*)

Saves an SMS message to device memory pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.3 int pack_sms_SendSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SendSMS_t * *reqParam*)

send sms list pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.4 int pack_sms_SetNewSMSCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SetNewSMSCallback_t *reqParam*)

set new sms callback pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.5 int pack_sms_SetSMSCAddress (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SetSMSCAddress_t * *reqParam*)

Set the SMS center address pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.6 int pack_sms_SLQSDeleteSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSDeleteSMS_t * *reqParam*)

delete sms pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.7 int pack_sms_SLQSGetIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets registration state of different WMS indications pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.8 int pack_sms_SLQSGetMessageWaiting (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the message waiting information pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.9 int pack_sms_SLQSGetSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSGetSMS_t * *reqParam*)

get sms pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.10 int pack_sms_SLQSGetSmsBroadcastConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSGetSmsBroadcastConfig_t * *reqParam*)

Provides Information about the SMS BroadcastConfiguration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.11 int pack_sms_SLQSGetSMSList (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSGetSMSList_t * *reqParam*)

get sms list pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.12 int pack_sms_SLQSGetTransLayerInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets information about the transport layer pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.13 int pack_sms_SLQSGetTransNWRegInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets transport layer network registration info pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.14 int pack_sms_SLQSModifySMSStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSModifySMSStatus_t * *reqParam*)

modify sms status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.15 int pack_sms_SLQSSendAsyncSMS (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSendAsyncSMS_t * *reqParam*)

Sends an SMS message for immediate over-the-air transmission pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.16 int pack_sms_SLQSSetIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetIndicationRegister_t * *reqParam*)

Sets the registration state of different WMS indications pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.17 int pack_sms_SLQSSetSmsBroadcastActivation (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetSmsBroadcastActivation_t * *reqParam*)

Enables or disables the reception of broadcast SMS messages pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.18 int pack_sms_SLQSSetSmsBroadcastConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetSmsBroadcastConfig_t * *reqParam*)

Sets the information about the SMS BroadcastConfiguration pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.19 int pack_sms_SLQSSetSmsStorage (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSetSmsStorage_t * *reqParam*)

Sets the SMS Storage on the device pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.20 int pack_sms_SLQSSmsGetMaxStorageSize (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSmsGetMaxStorageSize_t * *reqParam*)

Get the maximum number of messages that can be stored in the specified memory storage. Also it provides the number of slots currently available pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.21 int pack_sms_SLQSSmsGetMessageProtocol (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the message protocol currently in use for the WMS client pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.22 int pack_sms_SLQSSmsSetRoutes (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_sms_SLQSSmsSetRoutes_t * *reqParam*)

Sets the action performed on SMS message receipt for specified message routes. It also specifies the action performed on SMS receipt of status reports pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
	reqParam	packed request

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.23 int pack_sms_SLQSSwiGetSMSStorage (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the current SMS configuration that is applied to all incoming and outgoing messages pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.24 int unpack_sms_GetSMSCAddress (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_GetSMSCAddress_t * *pOutput*)

Gets the SMS center address unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.25 int unpack_sms_SaveSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SaveSMS_t * *pOutput*)

Saves an SMS message to device memory unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.26 int unpack_sms_SendSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SendSMS_t * *pOutput*)

send sms unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.27 int unpack_sms_SetNewSMSCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SetNewSMSCallback_t * *Output*)

set new sms callback unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*Output*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.28 int unpack_sms_SetNewSMSCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SetNewSMSCallback_ind_t * *pOutput*)

set new sms callback indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.29 int unpack_sms_SetSMSCAddress (uint8_t * *pResp*, uint16_t *respLen*)

Set the SMS center address unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.30 int unpack_sms_SLQSDeleteSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSDeleteSMS_t * *pOutput*)

delete sms unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.31 int unpack_sms_SLQSGetIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetIndicationRegister_t * *pOutput*)

Gets registration state of different WMS indications unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.32 int unpack_sms_SLQSGetMessageWaiting (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetMessageWaiting_t * *pOutput*)

Gets the message waiting information unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.33 int unpack_sms_SLQSGetSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetSMS_t * *pOutput*)

get sms unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.34 int unpack_sms_SLQSGetSmsBroadcastConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetSmsBroadcastConfig_t * *pOutput*)

Provides Information about the SMS BroadcastConfiguration unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.35 int unpack_sms_SLQSGetSMSList (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetSMSList_t * *pOutput*)

get sms list unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.36 int unpack_sms_SLQSGetTransLayerInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetTransLayerInfo_t * *pOutput*)

Gets information about the transport layer unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.37 int unpack_sms_SLQSGetTransNWRegInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetTransNWRegInfo_t * *pOutput*)

Gets transport layer network registration info unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.38 int unpack_sms_SLQSModifySMSStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSModifySMSStatus_t * *pOutput*)

modify sms status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.39 int unpack_sms_SLQSSendAsyncSMS (uint8_t * *pResp*, uint16_t *respLen*)

Sends an SMS message for immediate over-the-air transmission unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.40 int unpack_sms_SLQSSetIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*)

Sets the registration state of different WMS indications unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.41 int unpack_sms_SLQSSetSmsBroadcastActivation (uint8_t * *pResp*, uint16_t *respLen*)

Enables or disables the reception of broadcast SMS messages unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.42 int unpack_sms_SLQSSetSmsBroadcastConfig (uint8_t * *pResp*, uint16_t *respLen*)

Sets the information about the SMS BroadcastConfiguration unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.43 int unpack_sms_SLQSSetSmsStorage (uint8_t * *pResp*, uint16_t *respLen*)

Sets the SMS Storage on the device unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.44 int unpack_sms_SLQSSmsGetMaxStorageSize (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSmsGetMaxStorageSize_t * *pOutput*)

Get the maximum number of messages that can be stored in the specified memory storage. Also it provides the number of slots currently available unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.45 int unpack_sms_SLQSSmsGetMessageProtocol (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSmsGetMessageProtocol_t * *pOutput*)

Gets the message protocol currently in use for the WMS client unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.46 int unpack_sms_SLQSSmsSetRoutes (uint8_t * *pResp*, uint16_t *respLen*)

Sets the action performed on SMS message receipt for specified message routes. It also specifies the action performed on SMS receipt of status reports unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.47 int unpack_sms_SLQSSwiGetSMSStorage (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSSwiGetSMSStorage_t * *pOutput*)

Gets the current SMS configuration that is applied to all incoming and outgoing messages unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.29.4.48 int unpack_sms_SLQSWmsMemoryFullCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSWmsMemoryFullCallBack_ind_t * *pOutput*)

sms full callback indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30 swiavms.h File Reference

Data Structures

- struct pack_swiavms_SLQSAVMSStartSession_t
- struct unpack_swiavms_SLQSAVMSStartSession_t
- struct pack_swiavms_SLQSAVMSStopSession_t
- struct unpack_swiavms_SLQSAVMSStopSession_t
- struct _PackSwiAvmsSetSettingsPeriodInfo
- struct _PackSwiAvmsSetSettingsAPNInfo
- struct _PackSwiAvmsSetSettingsConnectionRetryTimers
- struct pack_swiavms_SLQSAVMSSetSettings_t
- struct unpack_swiavms_SLQSAVMSSetSettings_t
- struct _PackSwiAVMSSettingsAPNInfo
- struct _PackSwiAVMSSettingsConnectionRetryTimers
- struct _PackSwiAVMSSettingsPeriodsInfo
- struct unpack_swiavms_SLQSAVMSGetSettings_t
- struct pack_swiavms_SLQSAVMSSendSelection_t
- struct unpack_swiavms_SLQSAVMSSendSelection_t
- struct unpack_swiavms_SLQSAvmsSetEventReport_t
- struct _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo
- struct _UnpackSwiAvmsEventReportConfig
- struct _UnpackSwiAvmsEventReportNotification
- struct _UnpackSwiAvmsEventReportConnectionRequest
- struct _UnpackSwiAvmsEventReportWAMSParamChange
- struct _UnpackSwiAvmsEventReportPackageID
- struct _UnpackSwiAvmsEventReportRegStatus

- struct _UnpackSwiAvmsEventReportDataSessionStatus
- struct _UnpackSwiAvmsEventReportSessionType
- struct _UnpackSwiAvmsEventReportHTTPStatus
- struct _unpack_swiavms_SLQSAVMSSessionGetInfo_t
- struct _unpack_swiavms_SLQSAVMSEventReportInd_t

Macros

- #define LITEQMI_MAX_SWIOMA_STR_LEN 255
- #define MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH 49
- #define MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 29
- #define MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 29
- #define MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS 8
- #define MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER 8
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH 128
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH 128
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH 1024
- #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH 200

Typedefs

- typedef struct _PackSwiAvmsSetSettingsPeriodInfo PackSwiAvmsSetSettingsPeriodInfo
- typedef struct _PackSwiAvmsSetSettingsAPNInfo PackSwiAvmsSetSettingsAPNInfo
- typedef struct _PackSwiAvmsSetSettingsConnectionRetryTimers PackSwiAvmsSetSettingsConnectionRetryTimers
- typedef struct _PackSwiAVMSSettingsAPNInfo PackSwiAVMSSettingsAPNInfo
- typedef struct _PackSwiAVMSSettingsConnectionRetryTimers PackSwiAVMSSettingsConnectionRetryTimers
- typedef struct _PackSwiAVMSSettingsPeriodsInfo PackSwiAVMSSettingsPeriodsInfo
- typedef struct _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo UnpackSwiAvmsEventReportBinaryUpdateSessionInfo
- typedef struct _UnpackSwiAvmsEventReportConfig UnpackSwiAvmsEventReportConfig
- typedef struct _UnpackSwiAvmsEventReportNotification UnpackSwiAvmsEventReportNotification
- typedef struct _UnpackSwiAvmsEventReportConnectionRequest UnpackSwiAvmsEventReportConnectionRequest
- typedef struct _UnpackSwiAvmsEventReportWAMSParamChange UnpackSwiAvmsEventReportWAMSParamChange
- typedef struct _UnpackSwiAvmsEventReportPackageID UnpackSwiAvmsEventReportPackageID
- typedef struct _UnpackSwiAvmsEventReportRegStatus UnpackSwiAvmsEventReportRegStatus
- typedef struct _UnpackSwiAvmsEventReportDataSessionStatus UnpackSwiAvmsEventReportDataSessionStatus
- typedef struct _UnpackSwiAvmsEventReportSessionType UnpackSwiAvmsEventReportSessionType

- typedef struct _UnpackSwiAvmsEventReportHTTPStatus UnpackSwiAvmsEventReportHTTPStatus
- typedef struct _unpack_swiavms_SLQSAVMSSessionGetInfo_t unpack_swiavms_SLQSAVMSSessionGetInfo_t
- typedef struct _unpack_swiavms_SLQSAVMSEventReportInd_t unpack_swiavms_SLQSAVMSEventReportInd_t

Functions

- int pack_swiavms_SLQSAVMSStartSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSStartSession_t reqParam)
- int unpack_swiavms_SLQSAVMSStartSession (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSStartSession_t *pResponse)
- int pack_swiavms_SLQSAVMSStopSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSStopSession_t reqParam)
- int unpack_swiavms_SLQSAVMSStopSession (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSStopSession_t *pResponse)
- int pack_swiavms_SLQSAVMSSetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSSetSettings_t reqParam)
- int unpack_swiavms_SLQSAVMSSetSettings (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSSetSettings_t *pResponse)
- int pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSSetSettings_t reqParam)
- int pack_swiavms_SLQSAVMSGetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiavms_SLQSAVMSGetSettings (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSGetSettings_t *pResponse)
- int pack_swiavms_SLQSAVMSSendSelection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiavms_SLQSAVMSSendSelection_t reqParam)
- int unpack_swiavms_SLQSAVMSSendSelection (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSSendSelection_t *pResponse)
- int pack_swiavms_SLQSAvmsSetEventReport (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiavms_SLQSAvmsSetEventReport (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAvmsSetEventReport_t *pResponse)
- int pack_swiavms_SLQSAVMSSessionGetInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiavms_SLQSAVMSSessionGetInfo (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSSessionGetInfo_t *pResponse)
- int unpack_swiavms_SLQSAVMSEventReportInd (uint8_t *pResp, uint16_t respLen, unpack_swiavms_SLQSAVMSEventReportInd_t *pResponse)

9.30.1 Macro Definition Documentation

9.30.1.1 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.30.1.2 #define MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER 8

9.30.1.3 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH 1024

9.30.1.4 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH 128

9.30.1.5 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH 128

9.30.1.6 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH 200

9.30.1.7 #define MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH 49

9.30.1.8 #define MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS 8

9.30.1.9 #define MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 29

9.30.1.10 #define MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 29

9.30.2 Typedef Documentation

9.30.2.1 typedef struct _PackSwiAvmsSetSettingsAPNInfo PackSwiAvmsSetSettingsAPNInfo

Parameters

bAPNLength	APN Length.
szAPN	APN String.
bUnameLength	Username Length.
szUname	User Name String.
bPWDLength	Password Length.
szPWD	Password String.

9.30.2.2 typedef struct _PackSwiAvmsSetSettingsConnectionRetryTimers PackSwiAvmsSetSettingsConnectionRetryTimers

9.30.2.3 typedef struct _PackSwiAvmsSetSettingsPeriodInfo PackSwiAvmsSetSettingsPeriodInfo

Parameters

ulMin	Min period in seconds.
ulMax	Max period in seconds.

9.30.2.4 typedef struct _PackSwiAVMSSettingsAPNInfo PackSwiAVMSSettingsAPNInfo

9.30.2.5 typedef struct _PackSwiAVMSSettingsConnectionRetryTimers PackSwiAVMSSettingsConnectionRetryTimers

9.30.2.6 typedef struct _PackSwiAVMSSettingsPeriodsInfo PackSwiAVMSSettingsPeriodsInfo

9.30.2.7 typedef struct _unpack_swiavms_SLQSAVMSEventReportInd_t unpack_swiavms_SLQSAVMSEventReportInd_t

Structure that contains the session type response for AVMS event report indication command

Parameters

pBinaryUpdateSessionInfo[OUT]	• See UnpackSwiAvmsEventReportBinaryUpdateSessionInfo for more information
pNotification[OUT]	• See UnpackSwiAvmsEventReportConfig for more information
pPackageID[OUT]	• See UnpackSwiAvmsEventReportPackageID for more information
pConnectionRequest[OUT]	• See UnpackSwiAvmsEventReportConnectionRequest for more information

pWAMSParaChanged[OUT]	• See UnpackSwiAvmsEventReportWAMSParamChange for more information
pRegStatus[OUT]	• See UnpackSwiAvmsEventReportRegStatus for more information
pDataSessionStatus[OUT]	• See UnpackSwiAvmsEventReportDataSessionStatus for more information
pSessionType[OUT]	• See UnpackSwiAvmsEventReportSessionType for more information
pWAMSParaChanged[OUT]	• See UnpackSwiAvmsEventReportWAMSParamChange for more information
pHTTPStatus[OUT]	• See UnpackSwiAvmsEventReportHTTPStatus for more information
resultcode	- Tlv Result Code.

9.30.2.8 typedef struct _unpack_swiavms_SLQSAVMSSessionGetInfo_t unpack_swiavms_SLQSAVMSSessionGetInfo_t

Structure that contains the session type response for AVMS get session info command

Parameters

resultcode	- Tlv Result Code.

9.30.2.9 typedef struct _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo UnpackSwiAvmsEventReportBinaryUpdateSessionInfo

Parameters

bBinaryType	: Type • 1 - Firmware. • 2 - User App. • 3 - Legato Framework.
bStat	: State • 0x01 - No binary update available. • 0x02 - Query binary Download. • 0x03 - Binary Downloading. • 0x04 - Binary downloaded. • 0x05 -Query Binary Update. • 0x06 - Binary updating. • 0x07 - Binary updated.

bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
wUserInput-Timeout	Timeout for user input in minutes. A value of 0 means no time-out.
ulPkgDownload-Size	The size (in bytes) of the update package
ulPkgDownload-Complete	The number of bytes being downloaded. For downloading state, this value shall be $>$ 0 and incremented toward the pkg_dload_size. For other states, the value shall be 0 as it is meaningless.
wUpdate-CompeteStatus	Result code.This field should be looked at only when the AVMS session is complete.
bSerity	Serity. • 0x01 - Mandatory. • 0x02 - Optional.
wVersionLength	Length of FW Version string in bytes.
szVersion	FW Version string in ASCII.
wNameLength	Length Package Name string in bytes.
szName	Package Name in UCS2.
wDescription-Length	Length of description in bytes.
szDescription	Description of Update Package in USC2.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.10 typedef struct _UnpackSwiAvmsEventReportConfig UnpackSwiAvmsEventReportConfig

Parameters

bState	State. • 0x01 - AVMS Read Request. • 0x02 - AVMS Change Request. • 0x03 - AVMS Config Complete.
bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
wUserInput-Timeout	Timeout for user input in seconds. A value of 0 means no time-out
wAlertMsg-Length	Length of Alert message string in bytes.
szAlertMsg	Alert message in UCS2.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.11 typedef struct _UnpackSwiAvmsEventReportConnectionRequest UnpackSwiAvmsEventReport-ConnectionRequest

Parameters

bUserInput-Request	Bit mask of available user inputs. • 0x00 - No user input required. Informational indication. • 0x01 - Accept. • 0x02 - Reject.
bUserInput-Request	Timeout for user input in minutes. A value of 0 means no time-out.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.12 typedef struct _UnpackSwiAvmsEventReportDataSessionStatus UnpackSwiAvmsEventReportDataSessionStatus

Parameters

bType	Notification type. • 0: Data session closed. • 1: Data session activated. • 2: Register made. -3: Data session error.
wErrorCode	LWM2M Session error code. • 0x0000: none
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.13 typedef struct _UnpackSwiAvmsEventReportHTTPStatus UnpackSwiAvmsEventReportHTTPStatus

Parameters

wHTTPStatus	See RFC 7231. • 100 - Continue • 101 - Switching Protocols • 200 - OK • 201 - Created • 202 - Accepted • 203 - Non-Authoritative Information • 204 - No Content • 205 - Reset Content • 206 - Partial Content • 300 - Multiple Choices • 301 - Moved Permanently • 302 - Found • 303 - See Other • 304 - Not Modified • 305 - Use Proxy • 307 - Temporary Redirect • 400 - Bad Request • 401 - Unauthorized • 402 - Payment Required • 403 - Forbidden • 404 - Not Found • 405 - Method Not Allowed • 406 - Not Acceptable • 407 - Proxy Authentication Required • 408 - Request Timeout • 409 - Conflict • 410 - Gone • 411 - Length Required • 412 - Precondition Failed • 413 - Payload Too Large • 414 - URI Too Long • 415 - Unsupported Media Type • 416 - Range Not Satisfiable • 417 - Expectation Failed • 426 - Upgrade Required • 500 - Internal Server Error • 501 - Not Implemented • 502 - Bad Gateway • 503 - Service Unavailable • 504 - Gateway Timeout • 505 - HTTP Version Not Supported
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.14 typedef struct _UnpackSwiAvmsEventReportNotification UnpackSwiAvmsEventReportNotification

Parameters

bNotification	Notification. • 0x14 - Module starts sending data to server. • 0x15 - Authentication with the server. • 0x16 - session with the server is ended.
wSessionStatus	This field will set to the session status for notifications that occur at the end of a session, zero for all other notifications. • 0x0000: Successful: Session succeeded • 0x0001: Break: Session succeeded • 0x0002: Large Object Handled: Session succeeded • 0x0003: No status: Session succeeded • 0x0004: No more commands: Session succeeded • 0x0005: User cancel: Session cancelled • 0x0100-0x01FF: General errors • 0x0200-0x02FF: Syncml errors • 0x0300-0x03FF: Authentication errors • 0x0400-0x04FF: Protocol errors • 0x0500-0x05FF: Tree errors (DM Only) • 0x0600-0x06FF: Not applicable • 0x0700-0x07FF: Trigger errors • 0x0800-0x08FF: FUMO errors • 0x0900-0x09FF: Communication errors • 0x0A00-0x0AFF: Parsing errors • 0x0B00-0x0CFF: Not applicable • 0x7F00-0x7F12:Insignia errors • 0x7F13: Illegal text: Text received contains illegal characters • 0x7F14: Download failure: Failed to download FOTA image • 0x7F15: Empty session: Session ran successfully, but no information wasa updated • 0x7F16: Factory reset successful: Factory reset succeeded • 0x7F17: Factory reset fail: Factory reset failed
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.15 typedef struct _UnpackSwiAvmsEventReportPackageID UnpackSwiAvmsEventReportPackageID

Parameters

bPackageID	Package ID of the application binary that this AVMS_EVENT_ID notification is for.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.16 typedef struct _UnpackSwiAvmsEventReportRegStatus UnpackSwiAvmsEventReportRegStatus

Parameters

bRegStatus	LWM2M Registration status. • 0: Need Bootstrap. • 1: Bootstrap made. • 2: Register made. -3: Update made.
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.17 typedef struct _UnpackSwiAvmsEventReportSessionType UnpackSwiAvmsEventReportSessionType

Parameters

bType	Session Type. • 0: Bootstrap session. • 1. DM session
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.2.18 typedef struct _UnpackSwiAvmsEventReportWAMSParamChange UnpackSwiAvmsEventReportWAMSParamChange

Parameters

wWamsChangeMask	Mask of WAMS parameters changed. • By default set to 0xFF for all changes. • 0x01 - device_login • 0x02 - device_MD5_key • 0x04 - server_login • 0x08 - server_MD5_key • 0x10 - server_URL • 0x20 - Nonce • 0x40 - Application key
TlvPresent	Boolean indicating the presence of the TLV in the QMI response

9.30.3 Function Documentation

9.30.3.1 int pack_swiavms_SLQSAVMSGetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack AVMS Get settings command This maps to SLQSAVMSSetSettings

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.2 int pack_swiavms_SLQSAVMSSendSelection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSSendSelection_t *reqParam*)

Function to pack AVMS send selection command This maps to SLQSAVMSSendSelection

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSSendSelection_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.3 int pack_swiavms_SLQSAVMSSessionGetInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack AVMS get session info command This maps to SLQSAVMSSessionGetInfo

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes

in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.4 int pack_swiavms_SLQSAvmsSetEventReport (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack AVMS set event report command This maps to SLQSAVMSSetEventReport

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.5 int pack_swiavms_SLQSAVMSSetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSSetSettings_t *reqParam*)

Function to pack AVMS Set settings command This maps to SLQSAVMSSetSettings

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.6 int pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSSetSettings_t *reqParam*)

Function to pack AVMS Set settings without Auto Reobot field command This maps to SLQSAVMSSetSettings-IgnoreAutoReboot

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.7 int pack_swiavms_SLQSAVMSStartSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSStartSession_t *reqParam*)

Function to pack Start AVMS session command This maps to SLQSAVMSStartSession

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.8 int pack_swiavms_SLQSAVMSStopSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiavms_SLQSAVMSStopSession_t *reqParam*)

Function to pack cancel AVMS session command This maps to SLQSAVMSStopSession

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swiavms_SLQSAVMSStopSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 2 seconds

9.30.3.9 int unpack_swiavms_SLQSAVMSEventReportInd (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSEventReportInd_t * *pResponse*)

Function to unpack AVMS event report Indication command This maps to SLQSAVMSSetEventReport

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSEventReportInd_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.3.10 int unpack_swiavms_SLQSAVMSGetSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSGetSettings_t * *pResponse*)

Function to unpack AVMS set settings command This maps to SLQSAVMSGetSettings

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.3.11 int unpack_swiavms_SLQSAVMSSendSelection (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSSendSelection_t * *pResponse*)

Function to unpack AVMS send selection command This maps to SLQSAVMSSendSelection

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSendSelection_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.3.12 int unpack_swiavms_SLQSAVMSSessionGetInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSSessionGetInfo_t * *pResponse*)

Function to unpack AVMS event report Indication command This maps to SLQSAVMSSetEventReport

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSessionGetInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.3.13 int unpack_swiavms_SLQSAvmsSetEventReport (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAvmsSetEventReport_t * *pResponse*)

Function to unpack AVMS set event report command This maps to SLQSAVMSSetEventReport

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAvmsSetEventReport_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.3.14 int unpack_swiavms_SLQSAVMSSetSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSSetSettings_t * *pResponse*)

Function to unpack AVMS Set settings command This maps to SLQSAVMSSetSettings

Parameters

in	*pResp*	• Response from modem

in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.3.15 int unpack_swiavms_SLQSAVMSStartSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSStartSession_t * *pResponse*)

Function to unpack Start AVMS session response from modem This maps to SLQSAVMSStartSession

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pResponse*	• See unpack_swiavms_SLQSAVMSStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.3.16 int unpack_swiavms_SLQSAVMSStopSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiavms_SLQSAVMSStopSession_t * *pResponse*)

Function to pack stop AVMS session command This maps to SLQSAVMSStopSession

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

out	*pResponse*	• See unpack_swiavms_SLQSAVMSStopSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.31 SwiDataTypes.h File Reference

SWI data types.

Macros

- #define SWI_API
- #define QMI_NO_LTE_FW_SUPPORT 0
- #define QMI_TLV_PLACEHOLDER 0x8F
- #define UNUSEDPARAM(x) (void)x

Typedefs

- typedef unsigned int ULONG
- typedef unsigned long long ULONGLONG
- typedef signed char INT8
- typedef unsigned char BYTE
- typedef char CHAR
- typedef unsigned short WORD
- typedef unsigned short USHORT
- typedef const char * LPCSTR
- typedef int BOOL
- typedef signed short SHORT
- typedef signed int INT32
- typedef float FLOAT

9.31.1 Detailed Description

SWI data types.

9.31.2 Macro Definition Documentation

9.31.2.1 #define QMI_NO_LTE_FW_SUPPORT 0

9.31.2.2 #define QMI_TLV_PLACEHOLDER 0x8F

9.31.2.3 #define SWI_API

9.31.2.4 #define UNUSEDPARAM(x) (void)x

Macro used to avoid “unused variable” compiler warnings generated due to the inclusion of the "-Wextra" flag in our make files.

9.31.3 Typedef Documentation

9.31.3.1 typedef int BOOL

9.31.3.2 typedef unsigned char BYTE

9.31.3.3 typedef char CHAR

9.31.3.4 typedef float FLOAT

9.31.3.5 typedef signed int INT32

9.31.3.6 typedef signed char INT8

9.31.3.7 typedef const char* LPCSTR

9.31.3.8 typedef signed short SHORT

9.31.3.9 typedef unsigned int ULONG

9.31.3.10 typedef unsigned long long ULONGLONG

9.31.3.11 typedef unsigned short USHORT

9.31.3.12 typedef unsigned short WORD

9.32 swiloc.h File Reference

Data Structures

- struct unpack_swiloc_SwiLocGetAutoStart_t
- struct pack_swiloc_SwiLocSetAutoStart_t

Functions

- int pack_swiloc_SwiLocGetAutoStart (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swiloc_SwiLocGetAutoStart (uint8_t *pResp, uint16_t respLen, unpack_swiloc_SwiLocGetAutoStart_t *pOutput)
- int pack_swiloc_SwiLocSetAutoStart (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swiloc_SwiLocSetAutoStart_t *reqArg)
- int unpack_swiloc_SwiLocSetAutoStart (uint8_t *pResp, uint16_t respLen)

9.32.1 Function Documentation

9.32.1.1 int pack_swiloc_SwiLocGetAutoStart (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen)

Get Auto Start pack

Parameters

in	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.32.1.2 int pack_swiloc_SwiLocSetAutoStart (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swiloc_SwiLocSetAutoStart_t * *reqArg*)

Set Auto Start pack

Parameters

in	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.32.1.3 int unpack_swiloc_SwiLocGetAutoStart (uint8_t * *pResp*, uint16_t *respLen*, unpack_swiloc_SwiLocGetAutoStart_t * *pOutput*)

Get Auto Start unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.32.1.4 int unpack_swiloc_SwiLocSetAutoStart (uint8_t * *pResp*, uint16_t *respLen*)

Set Auto Start unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33 swioma.h File Reference

Data Structures

- struct pack_swioma_SLQSOMADMStartSession_t
- struct unpack_swioma_SLQSOMADMStartSession_t
- struct pack_swioma_SLQSOMADMCancelSession_t
- struct unpack_swioma_SLQSOMADMGetSettings_t
- struct pack_swioma_SLQSOMADMSetSettings_t
- struct pack_swioma_SLQSOMADMSendSelection_t
- struct pack_swioma_SLQSOMADMGetSessionInfo_t
- struct unpack_swioma_SLQSOMADMGetSessionInfo_t
- struct unpack_omaDmFotaTlv_t
- struct unpack_omaDmConfigTlv_t
- struct unpack_omaDmNotificationsTlv_t
- struct unpack_swioma_SLQSOMADMAlertCallback_ind_t

Macros

- #define LITEQMI_MAX_SWIOMA_STR_LEN 255

Functions

- int pack_swioma_SLQSOMADMStartSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMStartSession_t reqParam)
- int unpack_swioma_SLQSOMADMStartSession (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMStartSession_t *pOutput)
- int pack_swioma_SLQSOMADMCancelSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMCancelSession_t reqParam)
- int unpack_swioma_SLQSOMADMCancelSession (uint8_t *pResp, uint16_t respLen)
- int pack_swioma_SLQSOMADMGetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swioma_SLQSOMADMGetSettings (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMGetSettings_t *pOutput)
- int pack_swioma_SLQSOMADMSetSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSetSettings_t reqParam)
- int unpack_swioma_SLQSOMADMSetSettings (uint8_t *pResp, uint16_t respLen)

- int pack_swioma_SLQSOMADMSendSelection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMSendSelection_t reqParam)
- int unpack_swioma_SLQSOMADMSendSelection (uint8_t *pResp, uint16_t respLen)
- int pack_swioma_SLQSOMADMGetSessionInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_swioma_SLQSOMADMGetSessionInfo_t reqParam)
- int unpack_swioma_SLQSOMADMGetSessionInfo (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMGetSessionInfo_t *pOutput)
- int pack_swioma_SLQSOMADMAlertCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_swioma_SLQSOMADMAlertCallback (uint8_t *pResp, uint16_t respLen)
- int unpack_swioma_SLQSOMADMAlertCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_swioma_SLQSOMADMAlertCallback_ind_t *pOutput)

9.33.1 Macro Definition Documentation

9.33.1.1 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.33.2 Function Documentation

9.33.2.1 int pack_swioma_SLQSOMADMAlertCallback (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen*)

Function to pack QMI command to enable the SWIOMADM network-initiated alert callback function. This maps to SetSLQSOMADMAlertCallback

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.33.2.2 int pack_swioma_SLQSOMADMCancelSession (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* pack_swioma_SLQSOMADMCancelSession_t *reqParam*)

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSession

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On output, number of bytes actually packed
in	*reqParam*	• See pack_swioma_SLQSOMADMCancelSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.33.2.3 int pack_swioma_SLQSOMADMGetSessionInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMGetSessionInfo_t *reqParam*)

Function to pack QMI command to return information related to the current (or previous if no session is active) OMA-DM session. This maps to SLQSOMADMGetSessionInfo

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swioma_SLQSOMADMGetSessionInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.33.2.4 int pack_swioma_SLQSOMADMGetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Function to pack command to retrieve the OMA-DM settings from the device. This maps to SLQSOMADMGetSettings2

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On output, number of bytes actually packed

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.33.2.5 int pack_swioma_SLQSOMADMSendSelection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMSendSelection_t *reqParam*)

Function to pack OMA-DM send selection command This maps to SLQSOMADMSendSelection2

Parameters

out	*pCtx*	• See pack_qmi_t for more information

in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swioma_SLQSOMADMSendSelection_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.33.2.6 int pack_swioma_SLQSOMADMSetSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMSetSettings_t *reqParam*)

Function to pack OMA-DM set settings command This maps to SLQSOMADMSetSettings3

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swioma_SLQSOMADMSetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.33.2.7 int pack_swioma_SLQSOMADMStartSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_swioma_SLQSOMADMStartSession_t *reqParam*)

Function to pack Start OMA-DM session command This maps to SLQSOMADMStartSession2

Parameters

out	*pCtx*	• See pack_qmi_t for more information
in,out	*pReqBuf*	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
in,out	*pLen*	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
in	*reqParam*	• See pack_swioma_SLQSOMADMStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

Timeout: 20 seconds

9.33.2.8 int unpack_swioma_SLQSOMADMAlertCallback (uint8_t * *pResp*, uint16_t *respLen*)

Function to unpack response of QMI command to enable the SWIOMADM network-initiated alert callback function. This maps to SetSLQSOMADMAlertCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

- Please use eQMI_SWIOMA_EVENT_IND indication to identify this event from SWIOMA service read function

9.33.2.9 int unpack_swioma_SLQSOMADMAlertCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMAlertCallback_ind_t * *pOutput*)

Function to unpack SWIOMADM alert indications This maps to SetSLQSOMADMAlertCallback

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_swioma_SLQSOMADMAlertCallback_ind_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33.2.10 int unpack_swioma_SLQSOMADMCancelSession (uint8_t * *pResp*, uint16_t *respLen*)

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSession

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33.2.11 int unpack_swioma_SLQSOMADMGetSessionInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMGetSessionInfo_t * *pOutput*)

Function to unpack information related to the current (or previous if no session is active) OMA-DM session. This maps to SLQSOMADMGetSessionInfo

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_swioma_SLQSOMADMGetSessionInfo_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33.2.12 int unpack_swioma_SLQSOMADMGetSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMGetSettings_t * *pOutput*)

Function to unpack OMA-DM get settings response from modem This maps to SLQSOMADMGetSettings2

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem
out	*pOutput*	• See unpack_swioma_SLQSOMADMGetSettings_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33.2.13 int unpack_swioma_SLQSOMADMSendSelection (uint8_t * *pResp*, uint16_t *respLen*)

Function to unpack OMA-DM send selection command This maps to SLQSOMADMSendSelection2

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33.2.14 int unpack_swioma_SLQSOMADMSetSettings (uint8_t * *pResp*, uint16_t *respLen*)

Function to unpack OMA-DM set settings command This maps to SLQSOMADMSetSettings3

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33.2.15 int unpack_swioma_SLQSOMADMStartSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_swioma_SLQSOMADMStartSession_t * *pOutput*)

Function to unpack Start OMA-DM session response from modem This maps to SLQSOMADMStartSession2

Parameters

in	*pResp*	• Response from modem
in	*respLen*	• Length of pResp from modem

out	*pOutput*	• See unpack_swioma_SLQSOMADMStartSession_t for more information

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34 SWIWWANCMAPI.h File Reference

9.35 uim.h File Reference

Data Structures

- struct uim_appStatus
- struct uim_slotInfo
- struct uim_cardStatus
- struct uim_hotSwapStatus
- struct unpack_uim_GetCardStatus_t
- struct uim_encryptedPIN1
- struct uim_remainingRetries
- struct uim_sessionInformation
- struct uim_verifyUIMPIN
- struct uim_unblockUIMPIN
- struct uim_cardResult
- struct uim_setPINProtection
- struct uim_changeUIMPIN
- struct uim_fileInfo
- struct uim_UIMSessionInformation
- struct uim_readTransparentInfo
- struct uim_readResult
- struct pack_uim_VerifyPin_t
- struct unpack_uim_VerifyPin_t
- struct pack_uim_UnblockPin_t
- struct unpack_uim_UnblockPin_t
- struct pack_uim_SetPinProtection_t
- struct unpack_uim_SetPinProtection_t
- struct pack_uim_ChangePin_t
- struct unpack_uim_ChangePin_t
- struct pack_uim_ReadTransparent_t
- struct unpack_uim_ReadTransparent_t
- struct pack_uim_SLQSUIMEventRegister_t
- struct unpack_uim_SLQSUIMEventRegister_t
- struct appStats
- struct slotInf
- struct unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t
- struct slot_t
- struct slots_t

- struct unpack_uim_SLQSUIMGetSlotsStatus_t
- struct pack_uim_SLQSUIMSwitchSlot_t
- struct unpack_uim_SetUimSlotStatusChangeCallback_ind_t
- struct pack_uim_SLQSUIMPowerUp_t
- struct pack_uim_SLQSUIMPowerDown_t
- struct pack_uim_SLQSUIMRefreshOK_t
- struct uim_registerRefresh
- struct pack_uim_SLQSUIMRefreshRegister_t
- struct pack_uim_SLQSUIMRefreshComplete_t
- struct uim_refreshevent
- struct pack_uim_SLQSUIMRefreshGetLastEvent_t
- struct unpack_uim_SLQSUIMRefreshGetLastEvent_t
- struct pack_uim_SLQSUIMGetFileAttributes_t
- struct uim_fileAttributes
- struct unpack_uim_SLQSUIMGetFileAttributes_t
- struct uim_depersonalizationInformation
- struct pack_uim_SLQSUIMDepersonalization_t
- struct unpack_uim_SLQSUIMDepersonalization_t
- struct uim_authenticationData
- struct pack_uim_SLQSUIMAuthenticate_t
- struct uim_authenticateResult
- struct unpack_uim_SLQSUIMAuthenticate_t
- struct uim_personalizationStatus
- struct pack_uim_SLQSUIMGetConfiguration_t
- struct unpack_uim_SLQSUIMGetConfiguration_t

Macros

- #define UIM_UINT8_MAX_STRING_SZ 255
- #define UIM_MAX_DESCRIPTION_LENGTH 255
- #define UIM_MAX_NO_OF_SLOTS 5
- #define UIM_MAX_NO_OF_APPLICATIONS 10
- #define MAX_NO_OF_SLOTS 5
- #define MAX_NO_OF_APPLICATIONS 10
- #define MAX_DESCRIPTION_LENGTH 255
- #define MAX_SLOTS_STATUS 255
- #define MAX_ICCID_LENGTH 255
- #define UIM_MAX_CONTENT_LENGTH 1024
- #define UIM_MAX_ACTIVE_PERS_FEATURES 12

Functions

- int pack_uim_GetCardStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, void *reqArg)
- int unpack_uim_GetCardStatus (uint8_t *pResp, uint16_t respLen, unpack_uim_GetCardStatus_t *pOutput)
- int pack_uim_VerifyPin (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_VerifyPin_t *reqArg)
- int unpack_uim_VerifyPin (uint8_t *pResp, uint16_t respLen, unpack_uim_VerifyPin_t *pOutput)
- int pack_uim_UnblockPin (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_UnblockPin_t *reqArg)
- int unpack_uim_UnblockPin (uint8_t *pResp, uint16_t respLen, unpack_uim_UnblockPin_t *pOutput)
- int pack_uim_SetPinProtection (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SetPinProtection_t *reqArg)
- int unpack_uim_SetPinProtection (uint8_t *pResp, uint16_t respLen, unpack_uim_SetPinProtection_t *pOutput)

- int pack_uim_ChangePin (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_ChangePin_t *reqArg)
- int unpack_uim_ChangePin (uint8_t *pResp, uint16_t respLen, unpack_uim_ChangePin_t *pOutput)
- int pack_uim_ReadTransparent (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_ReadTransparent_t *reqArg)
- int unpack_uim_ReadTransparent (uint8_t *pResp, uint16_t respLen, unpack_uim_ReadTransparent_t *pOutput)
- int pack_uim_SLQSUIMEventRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMEventRegister_t *reqArg)
- int unpack_uim_SLQSUIMEventRegister (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMEventRegister_t *pOutput)
- int unpack_uim_SLQSUIMSetStatusChangeCallBack_ind (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t *pOutput)
- int pack_uim_SLQSUIMGetSlotsStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_uim_SLQSUIMGetSlotsStatus (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetSlotsStatus_t *pOutput)
- int pack_uim_SLQSUIMSwitchSlot (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMSwitchSlot_t *reqArg)
- int unpack_uim_SLQSUIMSwitchSlot (uint8_t *pResp, uint16_t respLen)
- int unpack_uim_SetUimSlotStatusChangeCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_uim_SetUimSlotStatusChangeCallback_ind_t *pOutput)
- int pack_uim_SLQSUIMPowerUp (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMPowerUp_t *reqArg)
- int unpack_uim_SLQSUIMPowerUp (uint8_t *pResp, uint16_t respLen)
- int pack_uim_SLQSUIMPowerDown (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMPowerDown_t *reqArg)
- int unpack_uim_SLQSUIMPowerDown (uint8_t *pResp, uint16_t respLen)
- int pack_uim_SLQSUIMReset (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_uim_SLQSUIMReset (uint8_t *pResp, uint16_t respLen)
- int pack_uim_SLQSUIMRefreshOK (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshOK_t *reqArg)
- int unpack_uim_SLQSUIMRefreshOK (uint8_t *pResp, uint16_t respLen)
- int pack_uim_SLQSUIMRefreshRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshRegister_t *reqArg)
- int unpack_uim_SLQSUIMRefreshRegister (uint8_t *pResp, uint16_t respLen)
- int pack_uim_SLQSUIMRefreshComplete (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshComplete_t *reqArg)
- int unpack_uim_SLQSUIMRefreshComplete (uint8_t *pResp, uint16_t respLen)
- int pack_uim_SLQSUIMRefreshGetLastEvent (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMRefreshGetLastEvent_t *reqArg)
- int unpack_uim_SLQSUIMRefreshGetLastEvent (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMRefreshGetLastEvent_t *pOutput)
- int pack_uim_SLQSUIMGetFileAttributes (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetFileAttributes_t *reqArg)
- int unpack_uim_SLQSUIMGetFileAttributes (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetFileAttributes_t *pOutput)
- int pack_uim_SLQSUIMDepersonalization (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMDepersonalization_t *reqArg)
- int unpack_uim_SLQSUIMDepersonalization (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMDepersonalization_t *pOutput)
- int pack_uim_SLQSUIMAuthenticate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMAuthenticate_t *reqArg)
- int unpack_uim_SLQSUIMAuthenticate (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMAuthenticate_t *pOutput)
- int pack_uim_SLQSUIMGetConfiguration (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_uim_SLQSUIMGetConfiguration_t *reqArg)
- int unpack_uim_SLQSUIMGetConfiguration (uint8_t *pResp, uint16_t respLen, unpack_uim_SLQSUIMGetConfiguration_t *pOutput)

9.35.1 Macro Definition Documentation

9.35.1.1 #define MAX_DESCRIPTION_LENGTH 255

9.35.1.2 #define MAX_ICCID_LENGTH 255

9.35.1.3 #define MAX_NO_OF_APPLICATIONS 10

9.35.1.4 #define MAX_NO_OF_SLOTS 5

9.35.1.5 #define MAX_SLOTS_STATUS 255

9.35.1.6 #define UIM_MAX_ACTIVE_PERS_FEATURES 12

9.35.1.7 #define UIM_MAX_CONTENT_LENGTH 1024

9.35.1.8 #define UIM_MAX_DESCRIPTION_LENGTH 255

9.35.1.9 #define UIM_MAX_NO_OF_APPLICATIONS 10

9.35.1.10 #define UIM_MAX_NO_OF_SLOTS 5

9.35.1.11 #define UIM_UINT8_MAX_STRING_SZ 255

9.35.2 Function Documentation

9.35.2.1 int pack_uim_ChangePin (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_ChangePin_t * *reqArg*)

Change Pin pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.2 int pack_uim_GetCardStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Card Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.3 int pack_uim_ReadTransparent (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_ReadTransparent_t * *reqArg*)

SLQS ReadTransparent pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.4 int pack_uim_SetPinProtection (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SetPinProtection_t * *reqArg*)

Set Pin Protection pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.5 int pack_uim_SLQSUIMAuthenticate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMAuthenticate_t * *reqArg*)

Send a security command to the card pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.6 int pack_uim_SLQSUIMDepersonalization (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMDepersonalization_t * *reqArg*)

De-activates or unblocks the personalization on the phone pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.7 int pack_uim_SLQSUIMEventRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMEventRegister_t * *reqArg*)

UIM Status Change callback enable pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.8 int pack_uim_SLQSUIMGetConfiguration (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMGetConfiguration_t * *reqArg*)

Gets the modem configuration for the UIM module pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.9 int pack_uim_SLQSUIMGetFileAttributes (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMGetFileAttributes_t * *reqArg*)

Retrieves the file attributes for any EF(Elementary File) or DF(Dedicated File) in the card and provides access by the path pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.10 int pack_uim_SLQSUIMGetSlotsStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.11 int pack_uim_SLQSUIMPowerDown (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMPowerDown_t * *reqArg*)

Powers down the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.12 int pack_uim_SLQSUIMPowerUp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMPowerUp_t * *reqArg*)

Powers up the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.13 int pack_uim_SLQSUIMRefreshComplete (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshComplete_t * *reqArg*)

Enables the terminal response to be sent to the card When the client finished Refresh procedure pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.14 int pack_uim_SLQSUIMRefreshGetLastEvent (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshGetLastEvent_t * *reqArg*)

Retrieve the last refresh event pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.15 int pack_uim_SLQSUIMRefreshOK (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshOK_t * *reqArg*)

Enables the client to indicate whether it is OK to start the Refresh procedure pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.16 int pack_uim_SLQSUIMRefreshRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMRefreshRegister_t * *reqArg*)

Registers for file change notifications triggered by the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.17 int pack_uim_SLQSUIMReset (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Resets the issuing control points state kept by the service pack.

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.18 int pack_uim_SLQSUIMSwitchSlot (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_SLQSUIMSwitchSlot_t * *reqArg*)

switch slot pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.19 int pack_uim_UnblockPin (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_UnblockPin_t * *reqArg*)

Unblock Pin pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.20 int pack_uim_VerifyPin (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_VerifyPin_t * *reqArg*)

Verify Pin Status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.21 int unpack_uim_ChangePin (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_ChangePin_t * *pOutput*)

Change Pin unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.22 int unpack_uim_GetCardStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_GetCardStatus_t * *pOutput*)

Get Card Status unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.23 int unpack_uim_ReadTransparent (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_ReadTransparent_t * *pOutput*)

SLQS ReadTransparent unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.24 int unpack_uim_SetPinProtection (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SetPinProtection_t * *pOutput*)

Set Pin Protection unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.25 int unpack_uim_SetUimSlotStatusChangeCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SetUimSlotStatusChangeCallback_ind_t * *pOutput*)

UIM Slot Status Change indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

use pack_uim_SLQSUIMEventRegister to subscribe

9.35.2.26 int unpack_uim_SLQSUIMAuthenticate (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMAuthenticate_t * *pOutput*)

Unpack the status code received from the card when card responded to the read request

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.27 int unpack_uim_SLQSUIMDepersonalization (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMDepersonalization_t * *pOutput*)

De-activates or unblocks the personalization on the phone unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.28 int unpack_uim_SLQSUIMEventRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMEvent-Register_t * *pOutput*)

UIM Status Change callback enable unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.29 int unpack_uim_SLQSUIMGetConfiguration (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetConfiguration_t * *pOutput*)

Gets the modem configuration for the UIM module unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.30 int unpack_uim_SLQSUIMGetFileAttributes (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetFileAttributes_t * *pOutput*)

Retrieves the file attributes for any EF(Elementary File) or DF(Dedicated File) in the card and provides access by the path unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.31 int unpack_uim_SLQSUIMGetSlotsStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMGetSlotsStatus_t * *pOutput*)

get slot status unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.32 int unpack_uim_SLQSUIMPowerDown (uint8_t * *pResp*, uint16_t *respLen*)

Powers down the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.33 int unpack_uim_SLQSUIMPowerUp (uint8_t * *pResp*, uint16_t *respLen*)

Powers up the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.34 int unpack_uim_SLQSUIMRefreshComplete (uint8_t * *pResp*, uint16_t *respLen*)

Enables the terminal response to be sent to the card When the client finished Refresh procedure unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.35 int unpack_uim_SLQSUIMRefreshGetLastEvent (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMRefreshGetLastEvent_t * *pOutput*)

Retrieve the last refresh event unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.36 int unpack_uim_SLQSUIMRefreshOK (uint8_t * *pResp*, uint16_t *respLen*)

Enables the client to indicate whether it is OK to start the Refresh procedure unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.37 int unpack_uim_SLQSUIMRefreshRegister (uint8_t * *pResp*, uint16_t *respLen*)

Registers for file change notifications triggered by the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.38 int unpack_uim_SLQSUIMReset (uint8_t * *pResp*, uint16_t *respLen*)

Resets the issuing control points state kept by the service unpack.

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.39 int unpack_uim_SLQSUIMSetStatusChangeCallBack_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t * *pOutput*)

UIM Status Change indication unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

use pack_uim_SLQSUIMEventRegister to subscribe

9.35.2.40 int unpack_uim_SLQSUIMSwitchSlot (uint8_t * *pResp*, uint16_t *respLen*)

switch slot unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.41 int unpack_uim_UnblockPin (uint8_t * *pResp*, uint16_t *respLen*, **unpack_uim_UnblockPin_t** * *pOutput*)

Unblock Pin unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.2.42 int unpack_uim_VerifyPin (uint8_t * *pResp,* uint16_t *respLen,* unpack_uim_VerifyPin_t * *pOutput*)

Verify Pin unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	qmi response length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36 voice.h File Reference

Data Structures

- struct voice_USSInfo
- struct pack_voice_OriginateUSSD_t
- struct pack_voice_AnswerUSSD_t
- struct voice_UUSInfo
- struct voice_CUGInfo
- struct voice_calledPartySubAdd
- struct pack_voice_SLQSVoiceDialCall_t
- struct voice_alphaIDInfo
- struct voice_ccSUPSType
- struct unpack_voice_SLQSVoiceDialCall_t
- struct pack_voice_SLQSVoiceEndCall_t
- struct unpack_voice_SLQSVoiceEndCall_t
- struct voice_callFwdTypeAndPlan
- struct pack_voice_SLQSVoiceSetSUPSService_t
- struct unpack_voice_SLQSVoiceSetSUPSService_t
- struct voice_airTimer
- struct voice_roamTimer
- struct voice_prefVoiceSO
- struct pack_voice_SLQSVoiceSetConfig_t
- struct unpack_voice_SLQSVoiceSetConfig_t
- struct pack_voice_SLQSVoiceAnswerCall_t
- struct unpack_voice_SLQSVoiceAnswerCall_t
- struct voice_CLIRResp
- struct unpack_voice_SLQSVoiceGetCLIR_t
- struct voice_CLIPResp
- struct unpack_voice_SLQSVoiceGetCLIP_t
- struct pack_voice_SLQSVoiceGetCallWaiting_t
- struct unpack_voice_SLQSVoiceGetCallWaiting_t
- struct pack_voice_SLQSVoiceGetCallBarring_t
- struct unpack_voice_SLQSVoiceGetCallBarring_t
- struct pack_voice_SLQSVoiceGetCallForwardingStatus_t
- struct voice_callFWInfo

- struct voice_callFWExtInfo
- struct voice_getCallFWInfo
- struct voice_getCallFWExtInfo
- struct unpack_voice_SLQSVoiceGetCallForwardingStatus_t
- struct pack_voice_SLQSVoiceSetCallBarringPassword_t
- struct unpack_voice_SLQSVoiceSetCallBarringPassword_t
- struct pack_voice_SLQSVoiceGetCallInfo_t
- struct voice_callInfo
- struct voice_remotePartyNum
- struct voice_remotePartyName
- struct voice_connectNumInfo
- struct voice_diagInfo
- struct unpack_voice_SLQSVoiceGetCallInfo_t
- struct voice_getAllCallInformation
- struct voice_getAllCallRmtPtyNum
- struct voice_getAllCallRmtPtyName
- struct voice_allCallsUUSInfo
- struct voice_allCallsAlphaIDInfo
- struct voice_allCallsDiagInfo
- struct voice_peerNumberInfo
- struct voice_arrCallInfo
- struct voice_arrRemotePartyNum
- struct voice_arrRemotePartyName
- struct voice_arrAlertingType
- struct voice_arrUUSInfo
- struct voice_arrSvcOption
- struct voice_arrCallEndReason
- struct voice_arrAlphaID
- struct voice_arrConnectPartyNum
- struct voice_arrDiagInfo
- struct voice_arrCalledPartyNum
- struct voice_arrRedirPartyNum
- struct voice_arrAlertingPattern
- struct unpack_voice_SLQSVoiceGetAllCallInfo_t
- struct pack_voice_SLQSVoiceManageCalls_t
- struct unpack_voice_SLQSVoiceManageCalls_t
- struct voice_burstDTMFInfo
- struct voice_DTMFLengths
- struct pack_voice_SLQSVoiceBurstDTMF_t
- struct unpack_voice_SLQSVoiceBurstDTMF_t
- struct pack_voice_SLQSVoiceStartContDTMF_t
- struct unpack_voice_SLQSVoiceStartContDTMF_t
- struct pack_voice_SLQSVoiceStopContDTMF_t
- struct unpack_voice_SLQSVoiceStopContDTMF_t
- struct pack_voice_SLQSVoiceSendFlash_t
- struct unpack_voice_SLQSVoiceSendFlash_t
- struct pack_voice_SLQSVoiceSetPreferredPrivacy_t
- struct pack_voice_SLQSVoiceIndicationRegister_t
- struct pack_voice_SLQSVoiceGetConfig_t
- struct voice_curAMRConfig
- struct unpack_voice_SLQSVoiceGetConfig_t
- struct pack_voice_SLQSVoiceOrigUSSDNoWait_t
- struct pack_voice_SLQSVoiceBindSubscription_t
- struct pack_voice_SLQSVoiceALSSetLineSwitching_t
- struct pack_voice_SLQSVoiceALSSelectLine_t

- struct voice_COLPResp
- struct unpack_voice_SLQSVoiceGetCOLP_t
- struct voice_COLRResp
- struct unpack_voice_SLQSVoiceGetCOLR_t
- struct voice_CNAPResp
- struct unpack_voice_SLQSVoiceGetCNAP_t
- struct pack_voice_SLQSOriginateUSSD_t
- struct unpack_voice_SLQSOriginateUSSD_t

Macros

- #define MAXVOICEUSSDLENGTH 182
- #define MAX_VOICE_CALL_NO_LEN 81
- #define MAX_VOICE_DESCRIPTION_LENGTH 255
- #define VOICE_MAX_NO_OF_CALLS 20
- #define BARRING_PASSWORD_LENGTH 4

Enumerations

- enum liteServiceClassInformation {
  LITE_VOICE_SUPS_SRV_CLASS_NONE = 0x00,
  LITE_VOICE_SUPS_SRV_CLASS_VOICE = 0x01,
  LITE_VOICE_SUPS_SRV_CLASS_DATA = 0x02,
  LITE_VOICE_SUPS_SRV_CLASS_FAX = 0x04,
  LITE_VOICE_SUPS_SRV_CLASS_SMS = 0x08,
  LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC = 0x10,
  LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC = 0x20,
  LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS = 0x40,
  LITE_VOICE_SUPS_SRV_CLASS_PADACCESS = 0x80 }

Functions

- int pack_voice_OriginateUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_OriginateUSSD_t *reqArg)
- int unpack_voice_OriginateUSSD (uint8_t *pResp, uint16_t respLen)
- int pack_voice_AnswerUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_AnswerUSSD_t *reqArg)
- int unpack_voice_AnswerUSSD (uint8_t *pResp, uint16_t respLen)
- int pack_voice_CancelUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_CancelUSSD (uint8_t *pResp, uint16_t respLen)
- int pack_voice_SLQSVoiceDialCall (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceDialCall_t *pReq)
- int unpack_voice_SLQSVoiceDialCall (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceDialCall_t *pOutput)
- int pack_voice_SLQSVoiceEndCall (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceEndCall_t *pReq)
- int unpack_voice_SLQSVoiceEndCall (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceEndCall_t *pOutput)
- int pack_voice_SLQSVoiceSetSUPSService (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetSUPSService_t *pReq)
- int unpack_voice_SLQSVoiceSetSUPSService (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSetSUPSService_t *pOutput)
- int pack_voice_SLQSVoiceSetConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetConfig_t *pReq)

- int unpack_voice_SLQSVoiceSetConfig (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSetConfig_t *pOutput)
- int pack_voice_SLQSVoiceAnswerCall (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceAnswerCall_t *pReq)
- int unpack_voice_SLQSVoiceAnswerCall (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceAnswerCall_t *pOutput)
- int pack_voice_SLQSVoiceGetCLIR (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCLIR (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCLIR_t *pOutput)
- int pack_voice_SLQSVoiceGetCLIP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCLIP (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCLIP_t *pOutput)
- int pack_voice_SLQSVoiceGetCallWaiting (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallWaiting_t *pReq)
- int unpack_voice_SLQSVoiceGetCallWaiting (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallWaiting_t *pOutput)
- int pack_voice_SLQSVoiceGetCallBarring (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallBarring_t *pReq)
- int unpack_voice_SLQSVoiceGetCallBarring (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallBarring_t *pOutput)
- int pack_voice_SLQSVoiceGetCallForwardingStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallForwardingStatus_t *pReq)
- int unpack_voice_SLQSVoiceGetCallForwardingStatus (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallForwardingStatus_t *pOutput)
- int pack_voice_SLQSVoiceSetCallBarringPassword (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetCallBarringPassword_t *pReq)
- int unpack_voice_SLQSVoiceSetCallBarringPassword (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSetCallBarringPassword_t *pOutput)
- int pack_voice_SLQSVoiceGetCallInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetCallInfo_t *pReq)
- int unpack_voice_SLQSVoiceGetCallInfo (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCallInfo_t *pOutput)
- int pack_voice_SLQSVoiceGetAllCallInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetAllCallInfo (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetAllCallInfo_t *pOutput)
- int pack_voice_SLQSVoiceManageCalls (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceManageCalls_t *pReq)
- int unpack_voice_SLQSVoiceManageCalls (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceManageCalls_t *pOutput)
- int pack_voice_SLQSVoiceBurstDTMF (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceBurstDTMF_t *pReq)
- int unpack_voice_SLQSVoiceBurstDTMF (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceBurstDTMF_t *pOutput)
- int pack_voice_SLQSVoiceStartContDTMF (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceStartContDTMF_t *pReq)
- int unpack_voice_SLQSVoiceStartContDTMF (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceStartContDTMF_t *pOutput)
- int pack_voice_SLQSVoiceStopContDTMF (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceStopContDTMF_t *pReq)
- int unpack_voice_SLQSVoiceStopContDTMF (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceStopContDTMF_t *pOutput)
- int pack_voice_SLQSVoiceSendFlash (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSendFlash_t *pReq)
- int unpack_voice_SLQSVoiceSendFlash (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceSendFlash_t *pOutput)
- int pack_voice_SLQSVoiceSetPreferredPrivacy (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceSetPreferredPrivacy_t *reqArg)

- int unpack_voice_SLQSVoiceSetPreferredPrivacy (uint8_t *pResp, uint16_t respLen)
- int pack_voice_SLQSVoiceIndicationRegister (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceIndicationRegister_t *reqArg)
- int unpack_voice_SLQSVoiceIndicationRegister (uint8_t *pResp, uint16_t respLen)
- int pack_voice_SLQSVoiceGetConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceGetConfig_t *reqArg)
- int unpack_voice_SLQSVoiceGetConfig (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetConfig_t *pOutput)
- int pack_voice_SLQSVoiceOrigUSSDNoWait (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceOrigUSSDNoWait_t *reqArg)
- int unpack_voice_SLQSVoiceOrigUSSDNoWait (uint8_t *pResp, uint16_t respLen)
- int pack_voice_SLQSVoiceBindSubscription (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceBindSubscription_t *reqArg)
- int unpack_voice_SLQSVoiceBindSubscription (uint8_t *pResp, uint16_t respLen)
- int pack_voice_SLQSVoiceALSSetLineSwitching (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceALSSetLineSwitching_t *reqArg)
- int unpack_voice_SLQSVoiceALSSetLineSwitching (uint8_t *pResp, uint16_t respLen)
- int pack_voice_SLQSVoiceALSSelectLine (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSVoiceALSSelectLine_t *reqArg)
- int unpack_voice_SLQSVoiceALSSelectLine (uint8_t *pResp, uint16_t respLen)
- int pack_voice_SLQSVoiceGetCOLP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCOLP (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCOLP_t *pOutput)
- int pack_voice_SLQSVoiceGetCOLR (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCOLR (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCOLR_t *pOutput)
- int pack_voice_SLQSVoiceGetCNAP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_voice_SLQSVoiceGetCNAP (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSVoiceGetCNAP_t *pOutput)
- int pack_voice_SLQSOriginateUSSD (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_voice_SLQSOriginateUSSD_t *reqArg)
- int unpack_voice_SLQSOriginateUSSD (uint8_t *pResp, uint16_t respLen, unpack_voice_SLQSOriginateUSSD_t *pOutput)

9.36.1 Macro Definition Documentation

9.36.1.1 #define BARRING_PASSWORD_LENGTH 4

9.36.1.2 #define MAX_VOICE_CALL_NO_LEN 81

9.36.1.3 #define MAX_VOICE_DESCRIPTION_LENGTH 255

9.36.1.4 #define MAXVOICEUSSDLENGTH 182

9.36.1.5 #define VOICE_MAX_NO_OF_CALLS 20

9.36.2 Enumeration Type Documentation

9.36.2.1 enum liteServiceClassInformation

Service Class information

Enumerator

LITE_VOICE_SUPS_SRV_CLASS_NONE

LITE_VOICE_SUPS_SRV_CLASS_VOICE

LITE_VOICE_SUPS_SRV_CLASS_DATA

LITE_VOICE_SUPS_SRV_CLASS_FAX

LITE_VOICE_SUPS_SRV_CLASS_SMS

LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC

LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC

LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS

LITE_VOICE_SUPS_SRV_CLASS_PADACCESS

9.36.3 Function Documentation

9.36.3.1 int pack_voice_AnswerUSSD (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_AnswerUSSD_t * *reqArg*)

Responds to a USSD request from the network pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.2 int pack_voice_CancelUSSD (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Cancels an in-progress USSD operation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.3 int pack_voice_OriginateUSSD (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_OriginateUSSD_t * *reqArg*)

Initiates a USSD operation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.4 int pack_voice_SLQSOriginateUSSD (**pack_qmi_t** * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, **pack_voice_SLQSOriginateUSSD_t** * *reqArg*)

Initiates a USSD session pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.5 int pack_voice_SLQSVoiceALSSelectLine (**pack_qmi_t** * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, **pack_voice_SLQSVoiceALSSelectLine_t** * *reqArg*)

allows the user to select the preferred line pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.6 int pack_voice_SLQSVoiceALSSetLineSwitching (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceALSSetLineSwitching_t * *reqArg*)

sets the line switch setting on the card pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.7 int pack_voice_SLQSVoiceAnswerCall (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceAnswerCall_t * *pReq*)

Answers an incoming voice call pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.8 int pack_voice_SLQSVoiceBindSubscription (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceBindSubscription_t * *reqArg*)

binds a subscription type to a specific voice client ID pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.9 int pack_voice_SLQSVoiceBurstDTMF (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceBurstDTMF_t * *pReq*)

Sends a burst Dual-Tone Multi frequency (DTMF) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.10 int pack_voice_SLQSVoiceDialCall (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceDialCall_t * *pReq*)

Originates a voice call (MO call) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.11 int pack_voice_SLQSVoiceEndCall (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceEndCall_t * *pReq*)

Ends a voice call pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.12 int pack_voice_SLQSVoiceGetAllCallInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

information associated with all the calls originating or terminating pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.13 int pack_voice_SLQSVoiceGetCallBarring (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceGetCallBarring_t * *pReq*)

Status of Call Barring Supplementary Service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.14 int pack_voice_SLQSVoiceGetCallForwardingStatus (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_voice_SLQSVoiceGetCallForwardingStatus_t * pReq)

Status of Call Forwarding Supplementary Service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.15 int pack_voice_SLQSVoiceGetCallInfo (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_voice_SLQSVoiceGetCallInfo_t * pReq)

information associated with a call pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.16 int pack_voice_SLQSVoiceGetCallWaiting (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_voice_SLQSVoiceGetCallWaiting_t * pReq)

Status of Call Waiting Supplementary Service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.17 int pack_voice_SLQSVoiceGetCLIP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

status of the Calling Line Identification Presentation (CLIP) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.18 int pack_voice_SLQSVoiceGetCLIR (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

status of the Calling Line Identification Restriction (CLIR) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.19 int pack_voice_SLQSVoiceGetCNAP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

status of the Calling Name Presentation(CNAP) pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.20 int pack_voice_SLQSVoiceGetCOLP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

status of the Connected Line Identification Presentation pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.21 int pack_voice_SLQSVoiceGetCOLR (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

status of the Connected Line Identification Restriction pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.22 int pack_voice_SLQSVoiceGetConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceGetConfig_t * *reqArg*)

retrieves various configuration parameters that control the modem behavior pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.23 int pack_voice_SLQSVoiceIndicationRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceIndicationRegister_t * *reqArg*)

Sets the registration state for different QMI_VOICE indications pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.24 int pack_voice_SLQSVoiceManageCalls (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceManageCalls_t * *pReq*)

Manages the calls by using the supplementary service pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.25 int pack_voice_SLQSVoiceOrigUSSDNoWait (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceOrigUSSDNoWait_t * *reqArg*)

initiates a USSD operation No Wait pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.26 int pack_voice_SLQSVoiceSendFlash (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSendFlash_t * *pReq*)

sends a simple flash message pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.27 int pack_voice_SLQSVoiceSetCallBarringPassword (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetCallBarringPassword_t * *pReq*)

Sets a Call Barring Password pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.28 int pack_voice_SLQSVoiceSetConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetConfig_t * *pReq*)

configuration parameters that control the modem behavior pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.29 int pack_voice_SLQSVoiceSetPreferredPrivacy (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetPreferredPrivacy_t * *reqArg*)

sets the voice privacy preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.30 int pack_voice_SLQSVoiceSetSUPSService (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceSetSUPSService_t * *pReq*)

call-independent supplementary services pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.31 int pack_voice_SLQSVoiceStartContDTMF (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceStartContDTMF_t * *pReq*)

Starts a continuous DTMF pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.32 int pack_voice_SLQSVoiceStopContDTMF (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_voice_SLQSVoiceStopContDTMF_t * *pReq*)

Stops a continuous DTMF pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.33 int unpack_voice_AnswerUSSD (uint8_t * *pResp*, uint16_t *respLen*)

Responds to a USSD request from the network unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.34 int unpack_voice_CancelUSSD (uint8_t * *pResp*, uint16_t *respLen*)

Cancels an in-progress USSD operation unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.35 int unpack_voice_OriginateUSSD (uint8_t * *pResp*, uint16_t *respLen*)

Initiates a USSD operation unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.36 int unpack_voice_SLQSOriginateUSSD (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSOriginateUSSD_t * *pOutput*)

Initiates a USSD session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.37 int unpack_voice_SLQSVoiceALSSelectLine (uint8_t * *pResp*, uint16_t *respLen*)

allows the user to select the preferred line unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.38 int unpack_voice_SLQSVoiceALSSetLineSwitching (uint8_t * *pResp*, uint16_t *respLen*)

sets the line switch setting on the card unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.39 int unpack_voice_SLQSVoiceAnswerCall (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceAnswerCall_t * *pOutput*)

Answers an incoming voice call unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.40 int unpack_voice_SLQSVoiceBindSubscription (uint8_t * *pResp*, uint16_t *respLen*)

binds a subscription type to a specific voice client ID unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.41 int unpack_voice_SLQSVoiceBurstDTMF (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceBurstDTMF_t * *pOutput*)

Sends a burst Dual-Tone Multi frequency (DTMF) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.42 int unpack_voice_SLQSVoiceDialCall (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceDialCall_t * *pOutput*)

Originates a voice call (MO call) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.43 int unpack_voice_SLQSVoiceEndCall (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceEndCall_t * *pOutput*)

Ends a voice call unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.44 int unpack_voice_SLQSVoiceGetAllCallInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetAllCallInfo_t * *pOutput*)

information associated with all the calls originating or terminating unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.45 int unpack_voice_SLQSVoiceGetCallBarring (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallBarring_t * *pOutput*)

Status of Call Barring Supplementary Service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.46 int unpack_voice_SLQSVoiceGetCallForwardingStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallForwardingStatus_t * *pOutput*)

Status of Call Forwarding Supplementary Service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.47 int unpack_voice_SLQSVoiceGetCallInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallInfo_t * *pOutput*)

information associated with a call unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.48 int unpack_voice_SLQSVoiceGetCallWaiting (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCallWaiting_t * *pOutput*)

Status of Call Waiting Supplementary Service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.49 int unpack_voice_SLQSVoiceGetCLIP (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCLIP_t * *pOutput*)

status of the Calling Line Identification Presentation (CLIP) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.50 int unpack_voice_SLQSVoiceGetCLIR (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCLIR_t * *pOutput*)

status of the Calling Line Identification Restriction (CLIR) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.51 int unpack_voice_SLQSVoiceGetCNAP (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCNAP_t * *pOutput*)

status of the Calling Name Presentation(CNAP) unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.52 int unpack_voice_SLQSVoiceGetCOLP (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCOLP_t * *pOutput*)

status of the Connected Line Identification Presentation unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.53 int unpack_voice_SLQSVoiceGetCOLR (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetCOLR_t * *pOutput*)

status of the Connected Line Identification Restriction unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.54 int unpack_voice_SLQSVoiceGetConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceGetConfig_t * *pOutput*)

retrieves various configuration parameters that control the modem behavior unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.55 int unpack_voice_SLQSVoiceIndicationRegister (uint8_t * *pResp*, uint16_t *respLen*)

Sets the registration state for different QMI_VOICE indications unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.56 int unpack_voice_SLQSVoiceManageCalls (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceManageCalls_t * *pOutput*)

Manages the calls by using the supplementary service unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.57 int unpack_voice_SLQSVoiceOrigUSSDNoWait (uint8_t * *pResp*, uint16_t *respLen*)

initiates a USSD operation No Wait unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.58 int unpack_voice_SLQSVoiceSendFlash (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSendFlash_t * *pOutput*)

sends a simple flash message unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.59 int unpack_voice_SLQSVoiceSetCallBarringPassword (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSetCallBarringPassword_t * *pOutput*)

Sets a Call Barring Password unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.60 int unpack_voice_SLQSVoiceSetConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSetConfig_t * *pOutput*)

configuration parameters that control the modem behavior unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.61 int unpack_voice_SLQSVoiceSetPreferredPrivacy (uint8_t * *pResp*, uint16_t *respLen*)

sets the voice privacy preference unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.62 int unpack_voice_SLQSVoiceSetSUPSService (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceSetSUPSService_t * *pOutput*)

call-independent supplementary services unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.63 int unpack_voice_SLQSVoiceStartContDTMF (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceStartContDTMF_t * *pOutput*)

Starts a continuous DTMF unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.36.3.64 int unpack_voice_SLQSVoiceStopContDTMF (uint8_t * *pResp*, uint16_t *respLen*, unpack_voice_SLQSVoiceStopContDTMF_t * *pOutput*)

Stops a continuous DTMF unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37 wds.h File Reference

Data Structures

- struct LibPackQosClassID
- struct LibPackTFTIDParams
- struct LibPackGPRSRequestedQoS
- struct LibPackUMTSQoS
- struct LibPackUMTSReqQoSSigInd
- struct pack_wds_SLQSStartDataSession_t
- struct unpack_wds_SLQSStartDataSession_t
- struct unpack_wds_SLQSSetPacketSrvStatusCallback_t
- struct pack_wds_SLQSStopDataSession_t
- struct wds_ProfileIdentifier
- struct wds_GPRSQoS
- struct wds_PCSCFIPv4ServerAddressList
- struct wds_PCSCFFQDNAddress
- struct wds_PCSCFFQDNAddressList
- struct wds_Domain
- struct wds_DomainNameList
- struct wds_IPV6AddressInfo
- struct wds_IPV6GWAddressInfo
- struct unpack_wds_SLQSGetRuntimeSettings_t
- struct wds_currNetworkInfo
- struct unpack_wds_SLQSSetWdsEventCallback_ind_t
- struct pack_wds_SLQSSetWdsEventCallback_t
- struct pack_wds_SLQSGetRuntimeSettings_t

- struct wds_UMTSMinQoS
- struct LibPackprofile_3GPP
- struct LibPackprofile_3GPP2
- union wds_profileInfo
- struct pack_wds_SLQSCreateProfile_t
- struct PackCreateProfileOut
- struct unpack_wds_SLQSCreateProfile_t
- struct pack_wds_SLQSModifyProfile_t
- struct unpack_wds_SLQSModifyProfile_t
- struct pack_wds_SLQSGetProfileSettings_t
- struct LibpackProfile3GPP
- struct LibpackProfile3GPP2
- union unpackWdsProfileParam
- struct UnPackGetProfileSettingOut
- struct unpack_wds_SLQSGetProfileSettings_t
- struct unpack_wds_GetSessionState_t
- struct pack_wds_GetDefaultProfile_t
- struct unpack_wds_GetDefaultProfile_t
- struct unpack_wds_GetConnectionRate_t
- struct pack_wds_GetPacketStatus_t
- struct unpack_wds_GetPacketStatus_t
- struct unpack_wds_GetSessionDuration_t
- struct pack_wds_GetSessionDuration_t
- struct unpack_wds_GetDormancyState_t
- struct pack_wds_GetDormancyState_t
- struct pack_wds_SLQSDeleteProfile_t
- struct unpack_wds_SLQSDeleteProfile_t
- struct pack_wds_SetDefaultProfile_t
- struct unpack_wds_SLQSGet3GPPConfigItem_t
- struct pack_wds_SLQSSet3GPPConfigItem_t
- struct unpack_wds_GetMobileIP_t
- struct pack_wds_GetMobileIP_t
- struct pack_wds_GetMobileIPProfile_t
- struct unpack_wds_GetMobileIPProfile_t
- struct currNetworkInfo
- struct unpack_wds_SLQSGetCurrDataSystemStat_t
- struct pack_wds_SLQSGetCurrDataSystemStat_t
- struct unpack_wds_GetLastMobileIPError_t
- struct pack_wds_GetLastMobileIPError_t
- struct rmTrasnferStaticsReq
- struct pack_wds_RMSetTransferStatistics_t
- struct unpack_wds_RMSetTransferStatistics_t
- struct pack_wds_SetMobileIPProfile_t
- struct unpack_wds_SetMobileIPProfile_t
- struct pack_wds_SLQSWdsSwiPDPRuntimeSettings_t
- struct ipv6AddressInfo
- struct unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t
- struct transferStatInd
- struct pack_wds_SLQSGetDUNCallInfo_t
- struct connectionStatus
- struct dunchannelRate
- struct unpack_wds_SLQSGetDUNCallInfo_t
- struct qmiWDSDataBearerTechnology
- struct unpack_wds_SLQSGetDataBearerTechnology_t
- struct pack_wds_SLQSGetDataBearerTechnology_t

- struct pack_wds_SLQSSetIPFamilyPreference_t
- struct unpack_wds_SLQSSetIPFamilyPreference_t
- struct pack_wds_SetDefaultProfileNum_t
- struct pack_wds_GetDefaultProfileNum_t
- struct unpack_wds_GetDefaultProfileNum_t
- struct wdsDhcpv4ProfileId
- struct wdsDhcpv4HwConfig
- struct wdsDhcpv4Option
- struct wdsDhcpv4OptionList
- struct pack_wds_SLQSSGetDHCPv4ClientConfig_t
- struct unpack_wds_SLQSSGetDHCPv4ClientConfig_t
- struct pack_wds_GetPacketStatistics_t
- struct unpack_wds_GetPacketStatistics_t
- struct unpack_wds_GetByteTotals_t
- struct unpack_wds_SLQSGetCurrentChannelRate_t
- struct unpack_wds_SLQSSGetLoopback_t
- struct pack_wds_SLQSSSetLoopback_t
- struct wds_DataULongTlv
- struct wds_DataULongLongTlv
- struct unpack_RMTransferStatistics_ind_t
- struct pack_wds_DHCPv4ClientLeaseChange_t
- struct wds_DHCPProfileIdTlv
- struct wds_DHCPLeaseStateTlv
- struct wds_IPv4AdTlv
- struct wds_DHCPOpt
- struct wds_DHCPLeaseOptTlv
- struct unpack_wds_DHCPv4ClientLease_ind_t
- struct pack_wds_SetMobileIP_t
- struct pack_wds_SetMobileIPParameters_t
- struct pack_wds_SetAutoconnect_t
- struct unpack_wds_GetAutoconnect_t
- struct wds_TrStatInd
- struct pack_wds_SLQSWdsSetEventReport_t
- struct wds_DHCPv4ProfileId
- struct wds_DHCPv4HWConfig
- struct wds_DHCPv4Option
- struct wds_DHCPv4OptionList
- struct pack_wds_SLQSSSetDHCPv4ClientConfig_t
- struct unpack_wds_GetDataBearerTechnology_t

Macros

- #define IPV6_ADDRESS_ARRAY_SIZE 8
- #define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24
- #define PACK_WDS_IPV4 4
- #define PACK_WDS_IPV6 6
- #define BYT_STAT_STAT_MASK 0X000000C0
- #define WDS_DHCP_MAX_NUM_OPTIONS 30
- #define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /∗ current max size of raw message in SDK process is 2048 ∗/

Typedefs

- typedef union unpackWdsProfileParam UnpackQmiProfileInfo

Enumerations

- enum liteQmiDataBearerMasks {
  QMI_LITE_WDS_CURRENT_CALL_DB_MASK = 0x01,
  QMI_LITE_WDS_LAST_CALL_DB_MASK = 0x02 }

Functions

- int pack_wds_SLQSStartDataSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSStartDataSession_t *reqArg)
- int unpack_wds_SLQSStartDataSession (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSStartDataSession_t *pOutput)
- int unpack_wds_SLQSSetPacketSrvStatusCallback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetPacketSrvStatusCallback_t *pOutput)
- int pack_wds_SLQSStopDataSession (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSStopDataSession_t *reqArg)
- int unpack_wds_SLQSStopDataSession (uint8_t *pResp, uint16_t respLen)
- int unpack_wds_SLQSGetRuntimeSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetRuntimeSettings_t *pOutput)
- int unpack_wds_SLQSSetWdsEventCallback_ind (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetWdsEventCallback_ind_t *pOutput)
- int unpack_wds_SLQSSetWdsEventCallback (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SLQSSetWdsEventCallback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetWdsEventCallback_t *reqArg)
- int pack_wds_SLQSGetRuntimeSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetRuntimeSettings_t *reqArg)
- int pack_wds_SLQSCreateProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSCreateProfile_t *reqArg)
- int unpack_wds_SLQSCreateProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSCreateProfile_t *pOutput)
- int pack_wds_SLQSModifyProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSModifyProfile_t *reqArg)
- int unpack_wds_SLQSModifyProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSModifyProfile_t *pOutput)
- int pack_wds_SLQSGetProfileSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetProfileSettings_t *reqArg)
- int unpack_wds_SLQSGetProfileSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetProfileSettings_t *pOutput)
- int pack_wds_GetSessionState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetSessionState (uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionState_t *pOutput)
- int pack_wds_GetDefaultProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfile_t *reqParam)
- int unpack_wds_GetDefaultProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfile_t *pOutput)
- int pack_wds_GetConnectionRate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetConnectionRate (uint8_t *pResp, uint16_t respLen, unpack_wds_GetConnectionRate_t *pOutput)
- int pack_wds_GetPacketStatus (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetPacketStatus_t *reqParam)
- int unpack_wds_GetPacketStatus (uint8_t *pResp, uint16_t respLen, unpack_wds_GetPacketStatus_t *pOutput)
- int pack_wds_GetSessionDuration (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetSessionDuration_t *reqParam)
- int unpack_wds_GetSessionDuration (uint8_t *pResp, uint16_t respLen, unpack_wds_GetSessionDuration_t *pOutput)

- int pack_wds_GetDormancyState (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDormancyState_t *reqParam)
- int unpack_wds_GetDormancyState (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDormancyState_t *pOutput)
- int pack_wds_SLQSDeleteProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSDeleteProfile_t *reqParam)
- int unpack_wds_SLQSDeleteProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSDeleteProfile_t *pOutput)
- int pack_wds_SetDefaultProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetDefaultProfile_t *reqParam)
- int unpack_wds_SetDefaultProfile (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SLQSGet3GPPConfigItem (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSGet3GPPConfigItem (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGet3GPPConfigItem_t *pOutput)
- int pack_wds_SLQSSet3GPPConfigItem (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSet3GPPConfigItem_t *reqParam)
- int unpack_wds_SLQSSet3GPPConfigItem (uint8_t *pResp, uint16_t respLen)
- int pack_wds_GetMobileIP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetMobileIP_t *pReqParam)
- int unpack_wds_GetMobileIP (uint8_t *pResp, uint16_t respLen, unpack_wds_GetMobileIP_t *pOutput)
- int pack_wds_GetMobileIPProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetMobileIPProfile_t *reqParam)
- int unpack_wds_GetMobileIPProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_GetMobileIPProfile_t *pOutput)
- int pack_wds_SLQSGetCurrDataSystemStat (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetCurrDataSystemStat_t *pReqParam)
- int unpack_wds_SLQSGetCurrDataSystemStat (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetCurrDataSystemStat_t *pOutput)
- int pack_wds_GetLastMobileIPError (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetLastMobileIPError_t *pReqParam)
- int unpack_wds_GetLastMobileIPError (uint8_t *pResp, uint16_t respLen, unpack_wds_GetLastMobileIPError_t *pOutput)
- int pack_wds_RMSetTransferStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_RMSetTransferStatistics_t *reqParam)
- int unpack_wds_RMSetTransferStatistics (uint8_t *pResp, uint16_t respLen, unpack_wds_RMSetTransferStatistics_t *pOutput)
- int pack_wds_SetMobileIPProfile (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIPProfile_t *reqParam)
- int unpack_wds_SetMobileIPProfile (uint8_t *pResp, uint16_t respLen, unpack_wds_SetMobileIPProfile_t *pOutput)
- int pack_wds_SLQSWdsSwiPDPRuntimeSettings (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSWdsSwiPDPRuntimeSettings_t *reqParam)
- int unpack_wds_SLQSWdsSwiPDPRuntimeSettings (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t *pOutput)
- int pack_wds_SLQSGetDUNCallInfo (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetDUNCallInfo_t *reqParam)
- int unpack_wds_SLQSGetDUNCallInfo (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetDUNCallInfo_t *pOutput)
- int pack_wds_SLQSGetDataBearerTechnology (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSGetDataBearerTechnology_t *pReqParam)
- int unpack_wds_SLQSGetDataBearerTechnology (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetDataBearerTechnology_t *pOutput)
- int pack_wds_SLQSSetIPFamilyPreference (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSetIPFamilyPreference_t *pReqParam)
- int unpack_wds_SLQSSetIPFamilyPreference (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSetIPFamilyPreference_t *pOutput)

- int pack_wds_SetDefaultProfileNum (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetDefaultProfileNum_t *pReqParam)
- int unpack_wds_SetDefaultProfileNum (uint8_t *pResp, uint16_t respLen)
- int pack_wds_GetDefaultProfileNum (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetDefaultProfileNum_t *pReqParam)
- int unpack_wds_GetDefaultProfileNum (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDefaultProfileNum_t *pOutput)
- int pack_wds_SLQSSGetDHCPv4ClientConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSGetDHCPv4ClientConfig_t *pReq)
- int unpack_wds_SLQSSGetDHCPv4ClientConfig (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSGetDHCPv4ClientConfig_t *pOutput)
- int pack_wds_GetPacketStatistics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_GetPacketStatistics_t *pReq)
- int unpack_wds_GetPacketStatistics (uint8_t *pResp, uint16_t respLen, unpack_wds_GetPacketStatistics_t *pOutput)
- int pack_wds_GetByteTotals (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetByteTotals (uint8_t *pResp, uint16_t respLen, unpack_wds_GetByteTotals_t *pOutput)
- int pack_wds_SLQSGetCurrentChannelRate (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSGetCurrentChannelRate (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSGetCurrentChannelRate_t *pOutput)
- int pack_wds_SLQSSGetLoopback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSSGetLoopback (uint8_t *pResp, uint16_t respLen, unpack_wds_SLQSSGetLoopback_t *pOutput)
- int pack_wds_SLQSSSetLoopback (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSSetLoopback_t *reqArg)
- int unpack_wds_SLQSSSetLoopback (uint8_t *pResp, uint16_t respLen)
- int unpack_wds_RMTransferStatistics_ind (uint8_t *pResp, uint16_t respLen, unpack_RMTransferStatistics_ind_t *pOutput)
- int pack_wds_DHCPv4ClientLeaseChange (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_DHCPv4ClientLeaseChange_t *reqArg)
- int unpack_wds_DHCPv4ClientLeaseChange (uint8_t *pResp, uint16_t respLen)
- int unpack_wds_DHCPv4ClientLease_ind (uint8_t *pResp, uint16_t respLen, unpack_wds_DHCPv4ClientLease_ind_t *pOutput)
- int pack_wds_SetMobileIP (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIP_t *reqArg)
- int unpack_wds_SetMobileIP (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SetMobileIPParameters (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetMobileIPParameters_t *reqArg)
- int unpack_wds_SetMobileIPParameters (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SetAutoconnect (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SetAutoconnect_t *reqArg)
- int unpack_wds_SetAutoconnect (uint8_t *pResp, uint16_t respLen)
- int pack_wds_GetAutoconnect (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetAutoconnect (uint8_t *pResp, uint16_t respLen, unpack_wds_GetAutoconnect_t *pOutput)
- int pack_wds_SLQSWdsSetEventReport (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSWdsSetEventReport_t *reqArg)
- int unpack_wds_SLQSWdsSetEventReport (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SLQSWdsGoDormant (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSWdsGoDormant (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SLQSWdsGoActive (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSWdsGoActive (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SLQSResetPacketStatics (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_SLQSResetPacketStatics (uint8_t *pResp, uint16_t respLen)
- int pack_wds_SLQSSSetDHCPv4ClientConfig (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen, pack_wds_SLQSSSetDHCPv4ClientConfig_t *reqArg)

- int unpack_wds_SLQSSSetDHCPv4ClientConfig (uint8_t *pResp, uint16_t respLen)
- int pack_wds_GetDataBearerTechnology (pack_qmi_t *pCtx, uint8_t *pReqBuf, uint16_t *pLen)
- int unpack_wds_GetDataBearerTechnology (uint8_t *pResp, uint16_t respLen, unpack_wds_GetDataBearerTechnology_t *pOutput)

9.37.1 Macro Definition Documentation

9.37.1.1 #define BYT_STAT_STAT_MASK 0X000000C0

9.37.1.2 #define IPV6_ADDRESS_ARRAY_SIZE 8

9.37.1.3 #define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24

9.37.1.4 #define PACK_WDS_IPV4 4

9.37.1.5 #define PACK_WDS_IPV6 6

9.37.1.6 #define WDS_DHCP_MAX_NUM_OPTIONS 30

9.37.1.7 #define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /* current max size of raw message in SDK process is 2048 */

9.37.2 Typedef Documentation

9.37.2.1 typedef union unpackWdsProfileParam UnpackQmiProfileInfo

This union WdsProfileParam consists of Profile3GPP and Profile3GPP2 out of which one will be used to create profile.

Parameters

SlqsProfile3GPP	3GPP profile
SlqsProfile3GP-P2	3GPP2 profile

9.37.3 Enumeration Type Documentation

9.37.3.1 enum liteQmiDataBearerMasks

Bit mask values to indicate the presence of data bearer information for the current and last data calls

Enumerator

QMI_LITE_WDS_CURRENT_CALL_DB_MASK

QMI_LITE_WDS_LAST_CALL_DB_MASK

9.37.4 Function Documentation

9.37.4.1 int pack_wds_DHCPv4ClientLeaseChange (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_DHCPv4ClientLeaseChange_t * *reqArg*)

DHCPv4 lease state changes pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.2 int pack_wds_GetAutoconnect (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets auto connect data session setting pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.3 int pack_wds_GetByteTotals (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get Rx/Tx byte counts since the start of the last packet data session pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.4 int pack_wds_GetConnectionRate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get connection rate pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.37.4.5 int pack_wds_GetDataBearerTechnology (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get current data bearer technology pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.6 int pack_wds_GetDefaultProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetDefaultProfile_t * *reqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.7 int pack_wds_GetDefaultProfileNum (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetDefaultProfileNum_t * *pReqParam*)

get default profile number pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.8 int pack_wds_GetDormancyState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetDormancyState_t * *reqParam*)

get dormancy state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.37.4.9 int pack_wds_GetLastMobileIPError (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetLastMobileIPError_t * *pReqParam*)

get current data system pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.37.4.10 int pack_wds_GetMobileIP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetMobileIP_t * *pReqParam*)

get mobile ip mode pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.37.4.11 int pack_wds_GetMobileIPProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetMobileIPProfile_t * *reqParam*)

get mobile ip profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.12 int pack_wds_GetPacketStatistics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetPacketStatistics_t * *pReq*)

gets current packet transfer counter values pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.13 int pack_wds_GetPacketStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetPacketStatus_t * *reqParam*)

get packet status pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.14 int pack_wds_GetSessionDuration (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_GetSessionDuration_t * *reqParam*)

get session duration pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.37.4.15 int pack_wds_GetSessionState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get session state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.16 int pack_wds_RMSetTransferStatistics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_RMSetTransferStatistics_t * *reqParam*)

rm set transfer statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Note

PDN Specific: No

9.37.4.17 int pack_wds_SetAutoconnect (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetAutoconnect_t * *reqArg*)

Auto connect data session parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.18 int pack_wds_SetDefaultProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetDefaultProfile_t * *reqParam*)

set default profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.19 int pack_wds_SetDefaultProfileNum (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetDefaultProfileNum_t * *pReqParam*)

set default profile number pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.20 int pack_wds_SetMobileIP (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetMobileIP_t * *reqArg*)

Sets the current mobile IP setting pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.21 int pack_wds_SetMobileIPParameters (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetMobileIPParameters_t * *reqArg*)

Sets the specified mobile IP parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.22 int pack_wds_SetMobileIPProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SetMobileIPProfile_t * *reqParam*)

set mobile ip profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.23 int pack_wds_SLQSCreateProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSCreateProfile_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.24 int pack_wds_SLQSDeleteProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSDeleteProfile_t * *reqParam*)

delete stored profile pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.25 int pack_wds_SLQSGet3GPPConfigItem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get 3Gpp config items pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.26 int pack_wds_SLQSGetCurrDataSystemStat (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetCurrDataSystemStat_t * *pReqParam*)

get current data system pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.27 int pack_wds_SLQSGetCurrentChannelRate (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get current Tx/Rx channel bitrate of the current packet data pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.28 int pack_wds_SLQSGetDataBearerTechnology (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetDataBearerTechnology_t * *pReqParam*)

get data bearer technology pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.29 int pack_wds_SLQSGetDUNCallInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetDUNCallInfo_t * *reqParam*)

get dun call info pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.30 int pack_wds_SLQSGetProfileSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetProfileSettings_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.31 int pack_wds_SLQSGetRuntimeSettings (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSGetRuntimeSettings_t * *reqArg*)

get runtime settings pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.32 int pack_wds_SLQSModifyProfile (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSModifyProfile_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.33 int pack_wds_SLQSResetPacketStatics (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Reset packet data transfer statistics pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.34 int pack_wds_SLQSSet3GPPConfigItem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSet3GPPConfigItem_t * *reqParam*)

set 3Gpp config items pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.35 int pack_wds_SLQSSetIPFamilyPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSetIPFamilyPreference_t * *pReqParam*)

Set IP Family Preference pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.36 int pack_wds_SLQSSetWdsEventCallback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSetWdsEventCallback_t * *reqArg*)

set event callback pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: No

9.37.4.37 int pack_wds_SLQSSGetDHCPv4ClientConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSGetDHCPv4ClientConfig_t * *pReq*)

get DHCPv4 Client Config pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*pReq*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.38 int pack_wds_SLQSSGetLoopback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

get the value of loopback mode and multiplier pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.39 int pack_wds_SLQSSSetDHCPv4ClientConfig (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSSetDHCPv4ClientConfig_t * *reqArg*)

Gets the DHCP Client V4 Configuration pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

see qmerrno.h for eQCWWAN_xxx error values

9.37.4.40 int pack_wds_SLQSSSetLoopback (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSSSetLoopback_t * *reqArg*)

Enable/disable Data Loopback Mode and set the value of loopback multiplier pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.41 int pack_wds_SLQSStartDataSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSStartDataSession_t * *reqArg*)

Start data session

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.42 int pack_wds_SLQSStopDataSession (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_wds_SLQSStopDataSession_t * *reqArg*)

stop data session pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.43 int pack_wds_SLQSWdsGoActive (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the device into Active state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.44 int pack_wds_SLQSWdsGoDormant (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Gets the device into dormant state pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.45 int pack_wds_SLQSWdsSetEventReport (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSWdsSetEventReport_t * reqArg)

Sets the event report parameters pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqArg*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.46 int pack_wds_SLQSWdsSwiPDPRuntimeSettings (pack_qmi_t * pCtx, uint8_t * pReqBuf, uint16_t * pLen, pack_wds_SLQSWdsSwiPDPRuntimeSettings_t * reqParam)

swi pdp runtime settings pack

Parameters

in,out	*pCtx*	qmi request context
out	*pReqBuf*	qmi request buffer
out	*pLen*	qmi request length
in	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Note

PDN Specific: Yes

9.37.4.47 int unpack_wds_DHCPv4ClientLease_ind (uint8_t * pResp, uint16_t respLen, unpack_wds_DHCPv4ClientLease_ind_t * pOutput)

DHCP lease state has changed indication unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.48 int unpack_wds_DHCPv4ClientLeaseChange (uint8_t * *pResp*, uint16_t *respLen*)

DHCPv4 lease state changes unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.49 int unpack_wds_GetAutoconnect (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetAutoconnect_t * *pOutput*)

Gets auto connect data session setting unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.50 int unpack_wds_GetByteTotals (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetByteTotals_t * *pOutput*)

get Rx/Tx byte counts since the start of the last packet data session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.51 int unpack_wds_GetConnectionRate (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetConnectionRate_t * *pOutput*)

get connection rate unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.52 int unpack_wds_GetDataBearerTechnology (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDataBearerTechnology_t * *pOutput*)

get current data bearer technology unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.53 int unpack_wds_GetDefaultProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDefaultProfile_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.54 int unpack_wds_GetDefaultProfileNum (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDefaultProfileNum_t * *pOutput*)

get default profile number unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.55 int unpack_wds_GetDormancyState (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDormancyState_t * *pOutput*)

get dormancy state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.56 int unpack_wds_GetLastMobileIPError (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetLastMobileIP-Error_t * *pOutput*)

get current data system unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.57 int unpack_wds_GetMobileIP (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetMobileIP_t * *pOutput*)

get mobile ip mode unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.58 int unpack_wds_GetMobileIPProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetMobileIPProfile_t * *pOutput*)

get mobile ip profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.59 int unpack_wds_GetPacketStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetPacketStatistics_t * *pOutput*)

gets current packet transfer counter values unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.60 int unpack_wds_GetPacketStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetPacketStatus_t * *pOutput*)

get packet status unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.61 int unpack_wds_GetSessionDuration (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetSessionDuration_t * *pOutput*)

get session duration unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.62 int unpack_wds_GetSessionState (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetSessionState_t * *pOutput*)

get session state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.63 int unpack_wds_RMSetTransferStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_RMSetTransferStatistics_t * *pOutput*)

rm set transfer statistics unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.64 int unpack_wds_RMTransferStatistics_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_RMTransferStatistics_ind_t * *pOutput*)

RM transfer statistics indication unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.65 int unpack_wds_SetAutoconnect (uint8_t * *pResp*, uint16_t *respLen*)

Auto connect data session parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.66 int unpack_wds_SetDefaultProfile (uint8_t * *pResp*, uint16_t *respLen*)

set default profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.67 int unpack_wds_SetDefaultProfileNum (uint8_t * *pResp*, uint16_t *respLen*)

set default profile number unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.68 int unpack_wds_SetMobileIP (uint8_t * *pResp*, uint16_t *respLen*)

Sets the current mobile IP setting unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.69 int unpack_wds_SetMobileIPParameters (uint8_t * *pResp*, uint16_t *respLen*)

Sets the specified mobile IP parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.70 int unpack_wds_SetMobileIPProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SetMobileIPProfile_t * *pOutput*)

set mobile ip profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.71 int unpack_wds_SLQSCreateProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSCreateProfile_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.72 int unpack_wds_SLQSDeleteProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSDeleteProfile_t * *pOutput*)

delete stored profile unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.73 int unpack_wds_SLQSGet3GPPConfigItem (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGet3GPPConfigItem_t * *pOutput*)

get 3GPP config items unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.74 int unpack_wds_SLQSGetCurrDataSystemStat (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetCurrDataSystemStat_t * *pOutput*)

get current data system unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.75 int unpack_wds_SLQSGetCurrentChannelRate (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetCurrentChannelRate_t * *pOutput*)

get current Tx/Rx channel bitrate of the current packet data unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.76 int unpack_wds_SLQSGetDataBearerTechnology (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetDataBearerTechnology_t * *pOutput*)

get data bearer technology unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.77 int unpack_wds_SLQSGetDUNCallInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetDUNCallInfo_t * *pOutput*)

get dun call info unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.78 int unpack_wds_SLQSGetProfileSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetProfileSettings_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.79 int unpack_wds_SLQSGetRuntimeSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSGetRuntimeSettings_t * *pOutput*)

get runtime settings unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.80 int unpack_wds_SLQSModifyProfile (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSModifyProfile_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response structure to fill

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.81 int unpack_wds_SLQSResetPacketStatics (uint8_t * *pResp*, uint16_t *respLen*)

Reset packet data transfer statistics unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.82 int unpack_wds_SLQSSet3GPPConfigItem (uint8_t * *pResp*, uint16_t *respLen*)

set 3GPP config items unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

9.37.4.83 int unpack_wds_SLQSSetIPFamilyPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSetIPFamilyPreference_t * *pOutput*)

Set IP Family Preference unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.84 int unpack_wds_SLQSSetPacketSrvStatusCallback (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSetPacketSrvStatusCallback_t * *pOutput*)

set packet srv status callback unpack

Parameters

in	*pResp*	qmi response
in	*respLen*	length
out	*pOutput*	unpacked response

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.85 int unpack_wds_SLQSSetWdsEventCallback (uint8_t * *pResp*, uint16_t *respLen*)

set event callback unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.86 int unpack_wds_SLQSSetWdsEventCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSetWdsEventCallback_ind_t * *pOutput*)

set event callback unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.87 int unpack_wds_SLQSSGetDHCPv4ClientConfig (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSGetDHCPv4ClientConfig_t * *pOutput*)

get DHCPv4 Client Config unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.88 int unpack_wds_SLQSSGetLoopback (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSSGetLoopback_t * *pOutput*)

get the value of loopback mode and multiplier unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.89 int unpack_wds_SLQSSetDHCPv4ClientConfig (uint8_t * *pResp*, uint16_t *respLen*)

Gets the DHCP Client V4 Configuration unpack.

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.90 int unpack_wds_SLQSSetLoopback (uint8_t * *pResp*, uint16_t *respLen*)

Enable/disable Data Loopback Mode and set the value of loopback multiplier unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.91 int unpack_wds_SLQSStartDataSession (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSStartDataSession_t * *pOutput*)

start data session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.92 int unpack_wds_SLQSStopDataSession (uint8_t * *pResp*, uint16_t *respLen*)

stop data session unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.93 int unpack_wds_SLQSWdsGoActive (uint8_t * *pResp*, uint16_t *respLen*)

Gets the device into Active state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.94 int unpack_wds_SLQSWdsGoDormant (uint8_t * *pResp*, uint16_t *respLen*)

Gets the device into dormant state unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.95 int unpack_wds_SLQSWdsSetEventReport (uint8_t * *pResp*, uint16_t *respLen*)

Sets the event report parameters unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.37.4.96 int unpack_wds_SLQSWdsSwiPDPRuntimeSettings (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t * *pOutput*)

get current data system unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOutput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

Index

_3gppRelease
  unpack_wds_SLQSGet3GPPConfigItem_t, 659
_PackSwiAVMSSettingsAPNInfo, 31
  bAPNLength, 31
  bPWDLength, 31
  bUnameLength, 31
  pAPN, 31
  pPWD, 31
  pUname, 31
_PackSwiAVMSSettingsConnectionRetryTimers, 31
  Timers, 32
_PackSwiAVMSSettingsPeriodsInfo, 32
  max, 32
  min, 32
_PackSwiAvmsSetSettingsAPNInfo, 30
  bAPNLength, 30
  bPWDLength, 30
  bUnameLength, 30
  szAPN, 30
  szPWD, 30
  szUname, 30
_PackSwiAvmsSetSettingsConnectionRetryTimers, 30
  Timers, 31
_PackSwiAvmsSetSettingsPeriodInfo, 31
  ulMax, 31
  ulMin, 31
_UnpackSwiAvmsEventReportBinaryUpdateSession-
    Info, 34
  bBinaryType, 35
  bSerity, 35
  bState, 35
  bUserInputRequest, 35
  szDescription, 35
  szName, 35
  szVersion, 36
  TlvPresent, 36
  ulPkgDownloadComplete, 36
  ulPkgDownloadSize, 36
  wDescriptionLength, 36
  wNameLength, 36
  wUpdateCompeteStatus, 36
  wUserInputTimeout, 36
  wVersionLength, 36
_UnpackSwiAvmsEventReportConfig, 36
  bState, 36
  bUserInputRequest, 37
  szAlertMsg, 37
  TlvPresent, 37
  wAlertMsgLength, 37
  wUserInputTimeout, 37
_UnpackSwiAvmsEventReportConnectionRequest, 37
  bUserInputRequest, 37
  TlvPresent, 37
  wUserInputTimeout, 37
_UnpackSwiAvmsEventReportDataSessionStatus, 37
  bType, 38
  TlvPresent, 38
  wErrorCode, 38
_UnpackSwiAvmsEventReportHTTPStatus, 38
  TlvPresent, 39
  wHTTPStatus, 39
_UnpackSwiAvmsEventReportNotification, 40
  bNotification, 40
  TlvPresent, 40
  wSessionStatus, 41
_UnpackSwiAvmsEventReportPackageID, 41
  bPackageID, 41
  TlvPresent, 41
_UnpackSwiAvmsEventReportRegStatus, 41
  bRegStatus, 41
  TlvPresent, 41
_UnpackSwiAvmsEventReportSessionType, 41
  bType, 42
  TlvPresent, 42
_UnpackSwiAvmsEventReportWAMSParamChange, 42
  TlvPresent, 42
  wWamsChangeMask, 42
_litefw_FirmwareInfo_, 29
  szCarrier_str, 29
  szCarrierPriversion_str, 30
  szFwversion_str, 30
  szModelid_str, 30
  szPackageid_str, 30
  szSku_str, 30
_unpack_swiavms_SLQSAVMSEventReportInd_t, 32
  pConfig, 33
  pNotification, 33
  resultcode, 33
_unpack_swiavms_SLQSAVMSSessionGetInfo_t, 33
  pConfig, 34
  pNotification, 34
  resultcode, 34

AAASPI
  unpack_wds_GetMobileIPProfile_t, 652
AAAState
  unpack_wds_GetMobileIPProfile_t, 652
ACT_CODE_MAX_SIZE
  dms.h, 746

ALS
voice_getAllCallInformation, 703
AMSSString
unpack_dms_GetFirmwareRevision_t, 451
unpack_dms_GetFirmwareRevisions_t, 452
APNName
unpack_wds_SLQSGetRuntimeSettings_t, 666
accelTemp
pack_loc_SLQSLOCInjectSensorData_t, 268
acceleroData
pack_loc_SLQSLOCInjectSensorData_t, 268
acceleroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 268
AccessMac
nas_protocolSubtypeElement, 186
accolc
pack_nas_SetACCOLC_t, 273
accuracy
pack_pds_SetPDSDefaults_t, 300
ackIndicator
sMSTransferRouteMTMessage, 410
acqOrdeLen
nas_acqOrderPref, 111
acroamsetting
pack_wds_SetAutoconnect_t, 363
acsetting
pack_wds_SetAutoconnect_t, 363
actCode
pack_dms_ActivateAutomatic_t, 241
ActSetCnt
nas_NetworkStat1x, 172
ActSetPilotPN
nas_ActPilotPNElement, 112
ActSetPilotPNStrength
nas_ActPilotPNElement, 112
action
pack_nas_InitiateDomainAttach_t, 273
activated_ind
sms_qaQmi3GPP2BroadcastCfgInfo, 400
sms_qaQmi3GPPBroadcastCfgInfo, 400
activationState
pack_dms_SetActivationStatusCallback_t, 242
activationStatus
dms_ActivationStatusTlv, 56
ActivationStatusTlv
unpack_dms_SetEventReport_ind_t, 464
activeBandClass
nas_RFInfoTlv, 191
RFBandInfoElements, 387
activeChannel
nas_RFInfoTlv, 191
RFBandInfoElements, 387
activeInd
sms_messageWaitingInfoContent, 398
ActiveStatus
voice_CLIPResp, 695
voice_CLIRResp, 696
voice_CNAPResp, 696
voice_COLPResp, 697
voice_COLRResp, 698
ActiveTechPref
unpack_nas_GetNetworkPreference_t, 512
addr
unpack_qos_IPv4Addr_t, 570
unpack_qos_IPv6Addr_t, 571
address
loc_IPv4Info, 102
loc_IPv6Info, 102
loc_urlAddr, 109
unpack_wds_GetMobileIPProfile_t, 652
addressSize
unpack_sms_GetSMSCAddress_t, 592
aid
uim_refreshevent, 430
uim_sessionInformation, 432
uim_UIMSessionInformation, 436
aidLength
appStats, 46
uim_appStatus, 415
uim_refreshevent, 430
uim_sessionInformation, 432
uim_UIMSessionInformation, 436
aidVal
appStats, 46
uim_appStatus, 415
aidingIndicatorMask
loc_sensorDataUsage, 107
airTimerValue
voice_airTimer, 676
alertingPattern
voice_arrAlertingPattern, 679
AlertingType
voice_arrAlertingType, 680
alertmsg
unpack_omaDmConfigTlv_t, 555
alertmsglength
unpack_omaDmConfigTlv_t, 555
allCallsAlphaIDInfoArr
voice_arrAlphaID, 681
AllCallsUUSInfo
voice_arrUUSInfo, 686
alphaDcs
voice_alphaIDInfo, 679
AlphaID
cat_AlPhaIdentifierTlv, 48
AlphaIDInfo
voice_allCallsAlphaIDInfo, 677
AlphaIDLength
cat_AlPhaIdentifierTlv, 48
alphaLen
voice_alphaIDInfo, 679
alphaText
voice_alphaIDInfo, 679
altSrcInfo_t, 42
coverage, 43
linkage, 43

source, 43
Altitude
unpack_pds_SLQSGetGPSStateInfo_t, 568
altitudeAssumed
unpack_loc_GnssSvInfo_Ind_t, 496
altitudeSrcInfo
pack_loc_SLQSLOCInjectPosition_t, 266
altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 266
altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 266
ambr_dl
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 576
ambr_dl_ext
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 576
ambr_dl_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 576
ambr_ul
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 576
ambr_ul_ext
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 576
ambr_ul_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 576
amssSize
unpack_dms_GetFirmwareRevision_t, 451
unpack_dms_GetFirmwareRevisions_t, 452
apdoxypages.c, 731
apnId
pack_qos_SLQSQosSwiReadApnExtraParams_t, 309
pack_qos_SLQSQosSwiReadDataStats_t, 310
unpack_qos_SLQSQosSwiReadApnExtraParams-_t, 576
unpack_qos_SLQSQosSwiReadDataStats_t, 577
apnName
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 674
apnname
unpack_wds_GetDefaultProfile_t, 649
apnsize
unpack_wds_GetDefaultProfile_t, 649
appNameLength
loc_LocApplicationInfo, 103
appProviderLength
loc_LocApplicationInfo, 103
appState
appStats, 46
uim_appStatus, 415
appStats, 43
aidLength, 46
aidVal, 46
appState, 46
appType, 46
persoFeature, 46
persoRetries, 46
persoState, 46
persoUnblockRetries, 46
pin1Retries, 46
pin1State, 46
pin2Retries, 46
pin2State, 46
puk1Retries, 46
puk2Retries, 46
univPin, 46
AppStatus
slotInf, 392
uim_slotInfo, 435
appType
appStats, 46
uim_appStatus, 416
appVersionLength
loc_LocApplicationInfo, 103
appVersionValid
loc_LocApplicationInfo, 103
Application
unpack_nas_GetCDMANetworkParameters_t, 508
appversion_str
unpack_dms_GetFirmwareInfo_t, 450
arfcn
nas_GERANInfo, 136
nas_gsmCellInfo, 137
arrfileInfo
uim_refreshevent, 430
uim_registerRefresh, 431
auth
unpack_wds_GetDefaultProfile_t, 649
authData
pack_uim_SLQSUIMAuthenticate_t, 332
AuthProt
nas_protocolSubtypeElement, 186
Authentication
unpack_wds_SLQSGetRuntimeSettings_t, 667
authentication
pack_wds_SetDefaultProfile_t, 365
AutoConnect
pack_swiavms_SLQSAVMSSetSettings_t, 322
unpack_swiavms_SLQSAVMSGetSettings_t, 601
AutoReboot
pack_swiavms_SLQSAVMSSetSettings_t, 322
unpack_swiavms_SLQSAVMSGetSettings_t, 601
Autosdm
unpack_swioma_SLQSOMADMGetSettings_t, 609
avgPeriod
nas_LTESigRptCfg, 161
nas_LTESigRptConfig, 162
azimuth
loc_satelliteInfo, 106

bAPNLength
_PackSwiAVMSSettingsAPNInfo, 31
_PackSwiAvmsSetSettingsAPNInfo, 30

bAuto
pack_pds_SetPortAutomaticTracking_t, 301
pack_pds_SetServiceAutomaticTracking_t, 301
bBinaryType
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 35
bEnable
pack_dms_UIMSetPINProtection_t, 253
pack_nas_SLQSSetSignalStrengthsCallback_t, 291
bEnabled
pack_pds_SetXTRAAutomaticDownload_t, 302
bForceDownload
pack_fms_SetImagesPreference_t, 257
bICCID
slot_t, 390
bICCIDLength
slot_t, 390
bLogicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 338
slot_t, 390
bNotification
_UnpackSwiAvmsEventReportNotification, 40
bNumberOfPhySlots
unpack_uim_SetUimSlotStatusChangeCallback_-ind_t, 613
BOOL
SwiDataTypes.h, 984
BPTlv
NASQmiCbkNasSystemSelPrefInd, 235
bPWDLength
_PackSwiAVMSSettingsAPNInfo, 31
_PackSwiAvmsSetSettingsAPNInfo, 30
bPackageID
_UnpackSwiAvmsEventReportPackageID, 41
bRegStatus
_UnpackSwiAvmsEventReportRegStatus, 41
bResetStatistics
rmTrasnferStaticsReq, 388
bSerity
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 35
bState
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 35
_UnpackSwiAvmsEventReportConfig, 36
bType
_UnpackSwiAvmsEventReportDataSessionStatus, 38
_UnpackSwiAvmsEventReportSessionType, 42
bUnameLength
_PackSwiAVMSSettingsAPNInfo, 31
_PackSwiAvmsSetSettingsAPNInfo, 30
bUserInputRequest
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 35
_UnpackSwiAvmsEventReportConfig, 37
_UnpackSwiAvmsEventReportConnectionRequest, 37
BYT_STAT_STAT_MASK
wds.h, 1046
BYTE
SwiDataTypes.h, 984
band
nas_LTEInfo, 153
band1900
nas_gsmCellInfo, 138
band_pref
NASBandPreferenceTlv, 225
BandCapability
unpack_dms_GetBandCapability_t, 440
bandCapability
unpack_dms_SLQSGetBandCapability_t, 471
bandPref
pack_nas_SLQSSetBandPreference_t, 290
bandwidth
nas_LTEInfo, 153
baseId
nas_CDMAInfo, 117
nas_CDMASysInfo, 120
baseLat
nas_CDMAInfo, 117
nas_CDMASysInfo, 120
baseLong
nas_CDMAInfo, 117
nas_CDMASysInfo, 121
BasestationID
unpack_nas_SLQSGetServingSystem_t, 528
BasestationLatitude
unpack_nas_SLQSGetServingSystem_t, 528
BasestationLongitude
unpack_nas_SLQSGetServingSystem_t, 528
BearerID
unpack_qos_QosFlowInfo_t, 574
bearerID
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 669
bearerId
unpack_QosFlowStat_t, 590
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 674
bootSize
unpack_dms_GetFirmwareRevisions_t, 452
BootString
unpack_dms_GetFirmwareRevisions_t, 452
bootversion_str
unpack_dms_GetFirmwareInfo_t, 450
Broadcast
unpack_nas_GetCDMANetworkParameters_t, 508
broadcastActivate
pack_sms_SLQSSetSmsBroadcastActivation_t, 318
broadcastConfig
sms_qaQmi3GPPBroadcastCfgInfo, 400
bsInfoValid

nas_CDMASysInfo, 121
bsPRev
nas_CDMASysInfo, 121
bsPRevValid
nas_CDMASysInfo, 121
bsic
nas_GERANInfo, 136
bsicId
nas_gsmCellInfo, 138
bucketSz
unpack_qos_tokenBucket_t, 588
buildID
FMSImageIdElement, 63
image_info_t, 67
buildIDLen
image_info_t, 67
buildIDLength
FMSImageIdElement, 63
buildId
FMSImageElement, 62
buildIdLength
FMSImageElement, 62
BurstDTMFInfo
pack_voice_SLQSVoiceBurstDTMF_t, 344
ByteLoopbackMode
unpack_wds_SLQSSGetLoopback_t, 672
ByteLoopbackMultiplier
unpack_wds_SLQSSGetLoopback_t, 672

CCETlv
unpack_cat_SetCatEventCallback_ind_t, 438
CDMA_P_Rev
unpack_nas_SLQSGetServingSystem_t, 528
CDMABroadcastConfig
sms_qaQmi3GPP2BroadcastCfgInfo, 400
CDMAECIOThreshListLen
nas_CDMAECIOThresh, 116
CDMARSSIThreshListLen
nas_CDMARSSIThresh, 117
CDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 542
CDMASystemInfoExt
unpack_nas_SLQSGetServingSystem_t, 528
CHAR
SwiDataTypes.h, 984
CK_MAX_SIZE
dms.h, 746
CQIValueCW0
unpack_nas_SLQSSwiGetLteCQI_t, 550
CQIValueCW1
unpack_nas_SLQSSwiGetLteCQI_t, 550
CSDomain
unpack_nas_GetServingNetwork_t, 515
CUGIndex
voice_CUGInfo, 700
CallBarStatus
unpack_nas_SLQSGetServingSystem_t, 528
callDuration
unpack_wds_GetSessionDuration_t, 656
callEndReason
unpack_wds_SLQSGetDUNCallInfo_t, 664
voice_arrCallEndReason, 682
CallFWExtInfo
voice_getCallFWExtInfo, 704
CallFWInfo
voice_getCallFWInfo, 705
callID
pack_voice_SLQSVoiceGetCallInfo_t, 348
pack_voice_SLQSVoiceStopContDTMF_t, 359
unpack_voice_SLQSVoiceStopContDTMF_t, 644
voice_allCallsAlphaIDInfo, 677
voice_allCallsDiagInfo, 677
voice_allCallsUUSInfo, 678
voice_arrAlertingPattern, 679
voice_arrAlertingType, 680
voice_arrCallEndReason, 682
voice_arrSvcOption, 686
voice_callInfo, 693
voice_getAllCallRmtPtyName, 703
voice_getAllCallRmtPtyNum, 704
voice_peerNumberInfo, 707
callNumber
pack_voice_SLQSVoiceDialCall_t, 345
callState
voice_callInfo, 693
callType
voice_callInfo, 694
CalledPartyNum
voice_arrCalledPartyNum, 681
callerID
voice_connectNumInfo, 699
callerIDLen
voice_connectNumInfo, 699
callerName
voice_remotePartyName, 709
Callinfo
voice_getAllCallInformation, 703
cardState
slotInf, 392
uim_slotInfo, 435
carrier
unpack_dms_SLQSSwiGetFirmwareCurr_t, 474
carrier_str
unpack_dms_GetFirmwareInfo_t, 450
CarrierImage_t, 46
m_FwBuildId, 47
m_FwImageId, 47
m_PriBuildId, 47
m_PriImageId, 47
m_nCarrierId, 47
m_nFolderId, 47
m_nStorage, 47
cat.h, 731
pack_cat_CATSendEnvelopeCommand, 732
pack_cat_CATSendTerminalResponse, 732
pack_cat_SetCATEventCallback, 733
unpack_cat_CATSendEnvelopeCommand, 733

unpack_cat_CATSendTerminalResponse, 733
unpack_cat_SetCATEventCallback, 734
unpack_cat_SetCatEventCallback_ind, 734
cat_AlPhaIdentifierTlv, 48
AlphaID, 48
AlphaIDLength, 48
ReferenceID, 48
cat_EndProactiveSessionTlv, 49
EndProactiveSession, 50
cat_EventIDDataTlv, 50
Data, 50
DataLength, 50
ReferenceID, 50
cat_EventListTlv, 50
SetupEventList, 51
cat_RefreshTlv, 51
RefreshMode, 51
RefreshStage, 51
cat_commonEventTlv, 48
CatEvent, 48
EventID, 49
EventLength, 49
TlvPresent, 49
cat_currentCatEvent, 49
CatAlphaIdtfr, 49
CatEndPS, 49
CatEvIDData, 49
CatEventLst, 49
CatRefresh, 49
CatAlphaIdtfr
cat_currentCatEvent, 49
CatEndPS
cat_currentCatEvent, 49
CatEvIDData
cat_currentCatEvent, 49
CatEvent
cat_commonEventTlv, 48
CatEventLst
cat_currentCatEvent, 49
CatRefresh
cat_currentCatEvent, 49
ccsSupported
nas_CDMASysInfo, 121
ccsSupportedValid
nas_CDMASysInfo, 121
cdmaSSInfo, 51
ecio, 52
rssi, 52
cdmaSysIdValid
nas_CDMASysInfo, 121
cell_resel_priority
nas_infoInterFreq, 148
cellBroadcastCap
nas_AddSysInfo, 113
CellID
unpack_nas_SLQSGetServingSystem_t, 528
cellID
nas_GERANInfo, 136
nas_UMTSInfo, 214
cellId
nas_GSMSysInfo, 141
nas_LTESysInfo, 166
nas_WCDMASysInfo, 223
cellIdValid
nas_gsmCellInfo, 138
nas_GSMSysInfo, 141
nas_LTESysInfo, 166
nas_WCDMASysInfo, 223
cellInterFreqParams
nas_infoInterFreq, 148
cellIsTDD
nas_umtsLTENbrCell, 216
CellParams
nas_LTEInfoIntrafreq, 156
cellReselPriority
nas_lteGsmCellInfo, 151
nas_LTEInfoIntrafreq, 156
nas_lteWcdmaCellInfo, 167
cells_len
nas_infoInterFreq, 148
nas_lteGsmCellInfo, 151
cellsLen
nas_LTEInfoIntrafreq, 156
nas_lteWcdmaCellInfo, 167
chaddr
wds_DHCPv4HWConfig, 718
wdsDhcpv4HwConfig, 729
chaddrLen
wds_DHCPv4HWConfig, 718
wdsDhcpv4HwConfig, 729
changePIN
pack_uim_ChangePin_t, 329
channelRate
unpack_wds_SLQSGetDUNCallInfo_t, 664
Chipset
nas_DeviceConfigDetail, 131
ckLen
uim_depersonalizationInformation, 421
ckVal
uim_depersonalizationInformation, 421
clear
pack_dms_SLQSSwiGetCrashInfo_t, 248
cmdID
pack_cat_CATSendEnvelopeCommand_t, 239
codingScheme
nas_PLMNNetworkNameData, 185
voice_remotePartyName, 710
common.h
eCAT, 737
eCTL, 737
eDMS, 737
eIMSA, 737
eIND, 738
eLOC, 737
eLOG_DEBUG, 737
eLOG_FATAL, 737

eLOG_INFO, 737
eLOG_WARN, 737
eNAS, 737
ePDS, 737
eQOS, 737
eREQ, 738
eRMS, 737
eRSP, 738
eSAR, 737
eSMS, 737
eSWILOC, 737
eSWIM2MCMD, 737
eSWIOMA, 737
eSWIOMAEXT, 737
eTIMEOUT_10_S, 738
eTIMEOUT_20_S, 738
eTIMEOUT_2_S, 738
eTIMEOUT_300_S, 738
eTIMEOUT_30_S, 738
eTIMEOUT_5_S, 738
eTIMEOUT_60_S, 738
eTIMEOUT_8_S, 738
eTIMEOUT_DEFAULT, 738
eTMD, 737
eUIM, 737
eVOICE, 737
eWDS, 737
common.h, 734
eLOG_LEVEL, 737
eQMI_SVC, 737
eTimeout, 737
fill_pack_ctx, 738
fill_sdu_hdr, 738
get_version, 738
glog, 739
gloglvl, 739
helper_get_resp_ctx, 738
helper_get_xid, 738
helper_set_log_func, 739
helper_set_log_lvl, 739
liteqmi_GetVersion, 739
liteqmi_helper_decode7bitAsciiEncString, 739
liteqmi_log, 739
logger, 737
MINREQBKLEN, 736
MSGID_AND_LEN, 736
MSGID_DONT_CARE, 736
msgtype, 738
SDU_HDR_LEN, 736
UNUSEDPARAM, 737
unpack_result_code_only, 739
commonInfo
unpack_nas_SLQSNasSwiModemStatus_t, 545
ConcSvcInfo
unpack_nas_SLQSGetServingSystem_t, 528
confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 270
conn_status
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 669
ConnectedPartyNum
voice_arrConnectPartyNum, 683
connectionStatus, 52
MDMCallDuration, 52
MDMConnStatus, 52
unpack_wds_GetSessionState_t, 657
unpack_wds_SLQSGetDUNCallInfo_t, 664
content
uim_authenticateResult, 416
uim_readResult, 428
contentLen
uim_authenticateResult, 416
uim_readResult, 428
context
uim_authenticationData, 417
contextId
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 383
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 674
contextType
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 383
ControlMac
nas_protocolSubtypeElement, 186
countryInitials
nas_PLMNNetworkNameData, 185
coverage
altSrcInfo_t, 43
cpich_ecno
nas_wcdmaCellInfo, 218
cpich_rscp
nas_wcdmaCellInfo, 218
crashAction
pack_dms_SetCrashAction_t, 243
crashData
crashInformation, 54
crashId
crashInformation, 54
crashInfo
crashInfoParams, 53
crashInfoParam
unpack_dms_SLQSSwiGetCrashInfo_t, 472
crashInfoParams, 53
crashInfo, 53
crashStatus, 53
crashInformation, 53
crashData, 54
crashId, 54
crashString, 54
crashStrlen, 54
gcdumpString, 54
gcdumpStrlen, 54
numCrashes, 54
crashStatus
crashInfoParams, 53

crashString
crashInformation, 54
crashStrlen
crashInformation, 54
csAttachState
nas_servSystem, 200
NASServingSystemInfo, 237
csBarStatus
nas_CallBarringSysInfo, 114
nas_callBarStatus, 115
cur_carr_name
unpack_dms_GetFirmwareInfo_t, 450
cur_carr_rev
unpack_dms_GetFirmwareInfo_t, 450
curDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 662
curProfile
pack_wds_SLQSModifyProfile_t, 375
UnPackGetProfileSettingOut, 675
CurrChanRxRate
dunchannelRate, 60
CurrChanTxRate
dunchannelRate, 60
currDBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
currNWInfo
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
currNetworkInfo, 54
NetworkType, 55
RATMask, 56
SOMask, 56
unpack_wds_SLQSGetCurrDataSystemStat_t, 660
current_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 661
current_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 661
currentChannelRXRate
unpack_wds_GetConnectionRate_t, 646
currentChannelTXRate
unpack_wds_GetConnectionRate_t, 646
currentDataBearer
pack_wds_SLQSSetWdsEventCallback_t, 377
currentNetwork
qmiWDSDataBearerTechnology, 386
CurrentPLMN
unpack_nas_SLQSGetServingSystem_t, 528
cust_attr
DMScustSettingInfo, 58
cust_id
DMScustSettingInfo, 58
DMSgetCustomInput, 60
pack_dms_GetCustFeaturesV2_t, 241
pack_dms_SetCustFeaturesV2_t, 246
cust_value
DMScustSettingInfo, 58
pack_dms_SetCustFeaturesV2_t, 246
custSetting
DMScustSettingList, 58
CustomSCP
unpack_nas_GetCDMANetworkParameters_t, 508

dBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
dBTechExtRatValue
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
dBTechExtSoMask
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
dBTechnology
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
dBtechExtAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
dBtechnologyExt
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
DHCPRelayEnabled
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
DHCPv4LeaseOptTlv
unpack_wds_DHCPv4ClientLease_ind_t, 645
DHCPv4LeaseStateTlv
unpack_wds_DHCPv4ClientLease_ind_t, 645
dLatitude
unpack_pds_SetEventReport_Ind_t, 563
dLongitude
unpack_pds_SetEventReport_Ind_t, 563
DMS_IMGDETAILS_LEN
dms.h, 746
DMS_PM_FACTORY
dms.h, 746
DMS_PM_LOW
dms.h, 746
DMS_PM_OFFLINE
dms.h, 746
DMS_PM_ONLINE
dms.h, 746
DMS_PM_RESET
dms.h, 746
DMS_PM_SHUT_DOWN
dms.h, 746
DMScustSettingInfo, 57
cust_attr, 58
cust_id, 58
cust_value, 58
id_length, 58
value_length, 58
DMScustSettingList, 58

custSetting, 58
list_type, 58
num_instances, 59
DMSgetCustomFeatureV2, 59
pCustSettingInfo, 59
pCustSettingList, 59
pGetCustomInput, 59
DMSgetCustomInput, 59
cust_id, 60
list_type, 60
DRCCover
nas_DRCParams, 132
DRCValue
nas_DRCParams, 132
DTMFInterdigitInterval
voice_DTMFLengths, 702
DTMFPulseWidth
voice_DTMFLengths, 702
DTMFdigit
pack_voice_SLQSVoiceStartContDTMF_t, 358
DTMInd
unpack_nas_SLQSGetServingSystem_t, 528
Data
cat_EventIDDataTlv, 50
data
sMSCAddress, 406
sMSEtwsMessage, 407
sMSTransferRouteMTMessage, 410
uim_authenticationData, 417
data_buf
NASOTAMessageTlv, 229
data_len
NASOTAMessageTlv, 229
dataBearer
pack_wds_SLQSSetWdsEventCallback_t, 378
dataBearerMask
unpack_wds_SLQSGetDataBearerTechnology_t, 662
dataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 664
dataBearerTechExt
pack_wds_SLQSSetWdsEventCallback_t, 378
dataCapabilities
nas_dataSrvCapabilities, 127
dataCapabilitiesLen
nas_dataSrvCapabilities, 127
DataCaps
unpack_nas_GetServingNetwork_t, 515
unpack_nas_GetServingNetworkCapabilities_t, 516
dataCaps
unpack_nas_SetDataCapabilitiesCallback_ind_t, 518
DataCapsLen
unpack_nas_GetServingNetwork_t, 515
unpack_nas_GetServingNetworkCapabilities_t, 516
dataCapsSize
unpack_nas_SetDataCapabilitiesCallback_ind_t, 518
dataLen
pack_cat_CATSendEnvelopeCommand_t, 239
pack_cat_CATSendTerminalResponse_t, 240
uim_authenticationData, 417
DataLength
cat_EventIDDataTlv, 50
DataRate
unpack_qos_swiQosFlow_t, 587
dataRateMax
unpack_qos_dataRate_t, 570
dataServiceCaCapability
unpack_dms_GetDeviceCapabilities_t, 446
DataServiceCapability
unpack_dms_GetDeviceCap_t, 445
DataSrvCapabilities
unpack_nas_SLQSGetServingSystem_t, 528
dataSysStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 670
dataSystemStatus
pack_wds_SLQSSetWdsEventCallback_t, 378
Date
unpack_swioma_SLQSOMADMGetSessionInfo_t, 607
DateLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 607
day
nas_timeInfo, 212
nas_UniversalTime, 218
dayLtSavingAdj
nas_timeInfo, 212
dayOfWeek
nas_timeInfo, 212
nas_UniversalTime, 218
daylightSavings
nas_qaQmi3Gpp2TimeZone, 187
defaultPDNEnabled
unpack_wds_SLQSGet3GPPConfigItem_t, 659
DefaultRoamInd
unpack_nas_SLQSGetServingSystem_t, 528
delayClass
LibPackGPRSRequestedQoS, 68
wds_GPRSQoS, 721
deliveryErrSDU
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
depersonilisationInfo
pack_uim_SLQSUIMDepersonalization_t, 333
description
unpack_omaDmFotaTlv_t, 557
descriptionlength
unpack_omaDmFotaTlv_t, 557
Desription
nas_QmiNas3GppNetworkInfo, 188
destPortRangeEnd

LibPackTFTIDParams, 92
destPortRangeStart
LibPackTFTIDParams, 93
DetailedSvcInfo
unpack_nas_SLQSGetServingSystem_t, 528
DevCrashState
unpack_dms_GetCrashAction_t, 441
DiagInfo
voice_allCallsDiagInfo, 677
voice_arrDiagInfo, 684
diagInfoLen
voice_diagInfo, 701
diagnosticInfo
voice_diagInfo, 701
digitCnt
voice_burstDTMFInfo, 687
dirNum
nas_dirNum, 131
dirNumLen
nas_dirNum, 132
direction
voice_callInfo, 694
DisableIMSI
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
displayCondition
nas_serviceProviderName, 199
dl_bw_value
nas_PhyCaAggPcellInfo, 177
nas_PhyCaAggScellDlBw, 178
nas_PhyCaAggScellInfo, 182
NASPhyCaAggPcellInfo, 230
NASPhyCaAggScellDlBw, 230
NASPhyCaAggScellInfo, 233
dms.h, 739
ACT_CODE_MAX_SIZE, 746
CK_MAX_SIZE, 746
DMS_IMGDETAILS_LEN, 746
DMS_PM_FACTORY, 746
DMS_PM_LOW, 746
DMS_PM_OFFLINE, 746
DMS_PM_ONLINE, 746
DMS_PM_RESET, 746
DMS_PM_SHUT_DOWN, 746
ERI_DATA_MAX_SIZE, 747
MAX_BUILD_ID_LEN, 747
MEID_MAX_SIZE, 747
pack_dms_ActivateAutomatic, 747
pack_dms_GetActivationState, 747
pack_dms_GetBandCapability, 747
pack_dms_GetCrashAction, 748
pack_dms_GetCustFeature, 748
pack_dms_GetCustFeaturesV2, 749
pack_dms_GetDeviceCap, 749
pack_dms_GetDeviceCapabilities, 749
pack_dms_GetDeviceHardwareRev, 750
pack_dms_GetDeviceMfr, 750
pack_dms_GetDeviceSerialNumbers, 750
pack_dms_GetFSN, 752
pack_dms_GetFirmwareInfo, 751
pack_dms_GetFirmwareRevision, 751
pack_dms_GetFirmwareRevisions, 752
pack_dms_GetHardwareRevision, 752
pack_dms_GetIMSI, 753
pack_dms_GetManufacturer, 753
pack_dms_GetModelID, 753
pack_dms_GetNetworkTime, 754
pack_dms_GetOfflineReason, 754
pack_dms_GetPRLVersion, 755
pack_dms_GetPower, 754
pack_dms_GetSerialNumbers, 755
pack_dms_GetUSBComp, 755
pack_dms_GetVoiceNumber, 756
pack_dms_ResetToFactoryDefaults, 756
pack_dms_SLQSDmsSwiGetResetInfo, 760
pack_dms_SLQSDmsSwiIndicationRegister, 760
pack_dms_SLQSGetBandCapability, 760
pack_dms_SLQSGetERIFile, 761
pack_dms_SLQSSwiClearDyingGaspStatistics, 761
pack_dms_SLQSSwiGetCrashInfo, 761
pack_dms_SLQSSwiGetDyingGaspCfg, 762
pack_dms_SLQSSwiGetDyingGaspStatistics, 762
pack_dms_SLQSSwiGetFirmwareCurr, 762
pack_dms_SLQSSwiGetFwUpdateStatus, 763
pack_dms_SLQSSwiGetHostDevInfo, 763
pack_dms_SLQSSwiGetOSInfo, 763
pack_dms_SLQSSwiGetSerialNoExt, 764
pack_dms_SLQSSwiSetDyingGaspCfg, 764
pack_dms_SLQSSwiSetHostDevInfo, 764
pack_dms_SLQSSwiSetOSInfo, 765
pack_dms_SLQSUIMGetState, 765
pack_dms_SetActivationStatusCallback, 757
pack_dms_SetCrashAction, 757
pack_dms_SetCustFeature, 757
pack_dms_SetCustFeaturesV2, 758
pack_dms_SetEventReport, 758
pack_dms_SetFirmwarePreference, 758
pack_dms_SetPower, 759
pack_dms_SetUSBComp, 759
pack_dms_UIMChangePIN, 765
pack_dms_UIMGetControlKeyStatus, 766
pack_dms_UIMGetICCID, 766
pack_dms_UIMGetPINStatus, 767
pack_dms_UIMSetControlKeyProtection, 767
pack_dms_UIMSetPINProtection, 767
pack_dms_UIMUnblockControlKey, 768
pack_dms_UIMUnblockPIN, 768
pack_dms_UIMVerifyPIN, 768
pack_dms_ValidateSPC, 769
SPC_SIZE, 747
UNIQUE_ID_LEN, 747
unpack_dms_ActivateAutomatic, 769
unpack_dms_GetActivationState, 770
unpack_dms_GetBandCapability, 770
unpack_dms_GetCrashAction, 770

unpack_dms_GetCustFeature, 771
unpack_dms_GetCustFeaturesV2, 771
unpack_dms_GetDeviceCap, 771
unpack_dms_GetDeviceCapabilities, 772
unpack_dms_GetDeviceHardwareRev, 772
unpack_dms_GetDeviceMfr, 772
unpack_dms_GetDeviceSerialNumbers, 773
unpack_dms_GetFSN, 774
unpack_dms_GetFirmwareInfo, 773
unpack_dms_GetFirmwareRevision, 773
unpack_dms_GetFirmwareRevisions, 774
unpack_dms_GetHardwareRevision, 774
unpack_dms_GetIMSI, 775
unpack_dms_GetManufacturer, 775
unpack_dms_GetModelID, 775
unpack_dms_GetNetworkTime, 776
unpack_dms_GetOfflineReason, 776
unpack_dms_GetPRLVersion, 777
unpack_dms_GetPower, 776
unpack_dms_GetSerialNumbers, 777
unpack_dms_GetUSBComp, 777
unpack_dms_GetVoiceNumber, 778
unpack_dms_ResetToFactoryDefaults, 778
unpack_dms_SLQSDmsSwiGetResetInfo, 782
unpack_dms_SLQSDmsSwiGetResetInfo_Ind, 782
unpack_dms_SLQSDmsSwiIndicationRegister, 782
unpack_dms_SLQSGetBandCapability, 783
unpack_dms_SLQSGetERIFile, 783
unpack_dms_SLQSSwiClearDyingGaspStatistics, 783
unpack_dms_SLQSSwiGetCrashInfo, 784
unpack_dms_SLQSSwiGetDyingGaspCfg, 784
unpack_dms_SLQSSwiGetDyingGaspStatistics, 784
unpack_dms_SLQSSwiGetFirmwareCurr, 785
unpack_dms_SLQSSwiGetFwUpdateStatus, 785
unpack_dms_SLQSSwiGetHostDevInfo, 786
unpack_dms_SLQSSwiGetOSInfo, 786
unpack_dms_SLQSSwiGetSerialNoExt, 786
unpack_dms_SLQSSwiSetDyingGaspCfg, 787
unpack_dms_SLQSSwiSetHostDevInfo, 787
unpack_dms_SLQSSwiSetOSInfo, 787
unpack_dms_SLQSUIMGetState, 788
unpack_dms_SetActivationStatusCallback, 778
unpack_dms_SetCrashAction, 779
unpack_dms_SetCustFeature, 779
unpack_dms_SetCustFeaturesV2, 779
unpack_dms_SetEventReport, 780
unpack_dms_SetEventReport_ind, 780
unpack_dms_SetFirmwarePreference, 780
unpack_dms_SetPower, 781
unpack_dms_SetUSBComp, 781
unpack_dms_UIMChangePIN, 788
unpack_dms_UIMGetControlKeyStatus, 788
unpack_dms_UIMGetICCID, 789
unpack_dms_UIMGetPINStatus, 789
unpack_dms_UIMSetControlKeyProtection, 789
unpack_dms_UIMSetPINProtection, 790
unpack_dms_UIMUnblockControlKey, 790
unpack_dms_UIMUnblockPIN, 790
unpack_dms_UIMVerifyPIN, 791
unpack_dms_ValidateSPC, 791
dms_ActivationStatusTlv, 56
activationStatus, 56
TlvPresent, 56
dms_OperatingModeTlv, 56
operatingMode, 57
TlvPresent, 57
domain
wds_DomainNameList, 720
domainLen
wds_Domain, 720
DomainList
unpack_wds_SLQSGetRuntimeSettings_t, 667
domainName
wds_Domain, 720
dormancyStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
dormancyState
unpack_wds_GetDormancyState_t, 650
dormancyStatus
pack_wds_SLQSSetWdsEventCallback_t, 378
unpack_wds_SLQSGetDUNCallInfo_t, 664
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
dtmSupp
nas_GSMSysInfo, 141
dtmSuppValid
nas_GSMSysInfo, 141
dunchannelRate, 60
CurrChanRxRate, 60
CurrChanTxRate, 60
MaxChanRxRate, 60
MaxChanTxRate, 60
Duration
pack_nas_SetNetworkPreference_t, 275
unpack_nas_GetNetworkPreference_t, 512

eCAT
common.h, 737
eCTL
common.h, 737
eDMS
common.h, 737
eFW_TYPE_CWE
lite-fw.h, 798
eFW_TYPE_CWE_NVU
lite-fw.h, 798
eFW_TYPE_INVALID
lite-fw.h, 798
eFW_TYPE_MBN
lite-fw.h, 798
eFW_TYPE_MBN_GOBI
lite-fw.h, 798

eFW_TYPE_NVU
lite-fw.h, 798
eFW_TYPE_SPK
lite-fw.h, 798
eIMSA
common.h, 737
eIND
common.h, 738
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED
nas.h, 828
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
nas.h, 828
eLOC
common.h, 737
eLOG_DEBUG
common.h, 737
eLOG_FATAL
common.h, 737
eLOG_INFO
common.h, 737
eLOG_WARN
common.h, 737
eModel_9X15
lite-fw.h, 799
eModel_9X30
lite-fw.h, 799
eModel_9x07
lite-fw.h, 799
eModel_Unknown
lite-fw.h, 799
eModel_WP9X15
lite-fw.h, 799
eNAS
common.h, 737
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100
nas.h, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15
nas.h, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25
nas.h, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50
nas.h, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6
nas.h, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75
nas.h, 828
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE
nas.h, 828
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE
nas.h, 828
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE
nas.h, 828
ePDS
common.h, 737
eQCWWAN_ERR_API_MUTEX_TIMEOUT
qmerrno.h, 927
eQCWWAN_ERR_BUFFER_SZ
qmerrno.h, 926
eQCWWAN_ERR_CANCEL_OP
qmerrno.h, 927
eQCWWAN_ERR_DRIVER
qmerrno.h, 927
eQCWWAN_ERR_ENUM_BEGIN
qmerrno.h, 926
eQCWWAN_ERR_ENUM_END
qmerrno.h, 927
eQCWWAN_ERR_FILE_COPY
qmerrno.h, 926
eQCWWAN_ERR_FILE_OPEN
qmerrno.h, 926
eQCWWAN_ERR_GENERAL
qmerrno.h, 926
eQCWWAN_ERR_INTERNAL
qmerrno.h, 926
eQCWWAN_ERR_INVALID_ARG
qmerrno.h, 926
eQCWWAN_ERR_INVALID_DEVID
qmerrno.h, 926
eQCWWAN_ERR_INVALID_FILE
qmerrno.h, 926
eQCWWAN_ERR_INVALID_QMI_RSP
qmerrno.h, 926
eQCWWAN_ERR_INVALID_XID
qmerrno.h, 927
eQCWWAN_ERR_MALFORMED_QMI_RSP
qmerrno.h, 926
eQCWWAN_ERR_MEMORY
qmerrno.h, 926
eQCWWAN_ERR_MULTIPLE_DEVICES
qmerrno.h, 927
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED
qmerrno.h, 927
eQCWWAN_ERR_NO_CANCELABLE_OP
qmerrno.h, 927
eQCWWAN_ERR_NO_CONNECTION
qmerrno.h, 926

eQCWWAN_ERR_NO_DEVICE
qmerrno.h, 926
eQCWWAN_ERR_NO_SIGNAL
qmerrno.h, 927
eQCWWAN_ERR_NONE
qmerrno.h, 926
eQCWWAN_ERR_NULL_TLV
qmerrno.h, 930
eQCWWAN_ERR_OFFLINE
qmerrno.h, 927
eQCWWAN_ERR_PDU_GENERATION
qmerrno.h, 927
eQCWWAN_ERR_QMI_ABORTED
qmerrno.h, 927
eQCWWAN_ERR_QMI_ACCESS_DENIED
qmerrno.h, 929
eQCWWAN_ERR_QMI_ACK_NOT_SENT
qmerrno.h, 929
eQCWWAN_ERR_QMI_ARG_TOO_LONG
qmerrno.h, 927
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED
qmerrno.h, 928
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK
qmerrno.h, 928
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED
qmerrno.h, 929
eQCWWAN_ERR_QMI_CALL_FAILED
qmerrno.h, 927
eQCWWAN_ERR_QMI_CARD_BUSY_RSP
qmerrno.h, 930
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED
qmerrno.h, 929
eQCWWAN_ERR_QMI_CAT_END
qmerrno.h, 930
eQCWWAN_ERR_QMI_CAT_START
qmerrno.h, 930
eQCWWAN_ERR_QMI_CAUSE_CODE
qmerrno.h, 928
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED
qmerrno.h, 927
eQCWWAN_ERR_QMI_CONNECT
qmerrno.h, 926
eQCWWAN_ERR_QMI_DEVICE_IN_USE
qmerrno.h, 927
eQCWWAN_ERR_QMI_DEVICE_NOT_READY
qmerrno.h, 928
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL
qmerrno.h, 928
eQCWWAN_ERR_QMI_DISABLED
qmerrno.h, 928
eQCWWAN_ERR_QMI_ENCODING
qmerrno.h, 928
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE
qmerrno.h, 930
eQCWWAN_ERR_QMI_EVENT_REG_FAILED
qmerrno.h, 930
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL
qmerrno.h, 929
eQCWWAN_ERR_QMI_FDN_RESTRICT
qmerrno.h, 929
eQCWWAN_ERR_QMI_FLOW_SUSPENDED
qmerrno.h, 928
eQCWWAN_ERR_QMI_GENERAL
qmerrno.h, 928
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED
qmerrno.h, 929
eQCWWAN_ERR_QMI_IFACE
qmerrno.h, 926
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE
qmerrno.h, 929
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER
qmerrno.h, 928
eQCWWAN_ERR_QMI_INCORRECT_PIN
qmerrno.h, 927
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE
qmerrno.h, 929
eQCWWAN_ERR_QMI_INJECT_TIMEOUT
qmerrno.h, 929
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES
qmerrno.h, 928
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND
qmerrno.h, 928
eQCWWAN_ERR_QMI_INTERNAL
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_ARG
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD
qmerrno.h, 930
eQCWWAN_ERR_QMI_INVALID_HANDLE
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_ID
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_INDEX
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_OPERATION
qmerrno.h, 929
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_PINID
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_PROFILE
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE

qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_QMI_CMD
qmerrno.h, 929
eQCWWAN_ERR_QMI_INVALID_QOS_ID
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_TECH_PREF
qmerrno.h, 927
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP
qmerrno.h, 930
eQCWWAN_ERR_QMI_INVALID_TRANSITION
qmerrno.h, 928
eQCWWAN_ERR_QMI_INVALID_TX_ID
qmerrno.h, 927
eQCWWAN_ERR_QMI_MALFORMED_MSG
qmerrno.h, 927
eQCWWAN_ERR_QMI_MAX
qmerrno.h, 929
eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE
qmerrno.h, 928
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE
qmerrno.h, 928
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE
qmerrno.h, 928
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT
qmerrno.h, 928
eQCWWAN_ERR_QMI_MISSING_ARG
qmerrno.h, 927
eQCWWAN_ERR_QMI_MSG_BLOCKED
qmerrno.h, 929
eQCWWAN_ERR_QMI_NETWORK_ABORTED
qmerrno.h, 929
eQCWWAN_ERR_QMI_NETWORK_NOT_READY
qmerrno.h, 928
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE
qmerrno.h, 928
eQCWWAN_ERR_QMI_NO_EFFECT
qmerrno.h, 927
eQCWWAN_ERR_QMI_NO_ENTRY
qmerrno.h, 928
eQCWWAN_ERR_QMI_NO_FREE_PROFILE
qmerrno.h, 927
eQCWWAN_ERR_QMI_NO_MEMORY
qmerrno.h, 927
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND
qmerrno.h, 927
eQCWWAN_ERR_QMI_NO_RADIO
qmerrno.h, 929
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION
qmerrno.h, 929
eQCWWAN_ERR_QMI_NO_THRESHOLDS
qmerrno.h, 927
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE
qmerrno.h, 928
eQCWWAN_ERR_QMI_NOT_PROVISIONED
qmerrno.h, 927
eQCWWAN_ERR_QMI_NOT_SUPPORTED
qmerrno.h, 929
eQCWWAN_ERR_QMI_OFFSET
qmerrno.h, 927
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED
qmerrno.h, 927
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED
qmerrno.h, 927
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE
qmerrno.h, 929
eQCWWAN_ERR_QMI_OUT_OF_CALL
qmerrno.h, 927
eQCWWAN_ERR_QMI_PIN_BLOCKED
qmerrno.h, 928
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED
qmerrno.h, 928
eQCWWAN_ERR_QMI_POLICY_MISMATCH
qmerrno.h, 929
eQCWWAN_ERR_QMI_REQ
qmerrno.h, 926
eQCWWAN_ERR_QMI_REQ_SCH
qmerrno.h, 926
eQCWWAN_ERR_QMI_REQ_TO
qmerrno.h, 926
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED
qmerrno.h, 928
eQCWWAN_ERR_QMI_RSP
qmerrno.h, 926
eQCWWAN_ERR_QMI_RSP_TO
qmerrno.h, 926
eQCWWAN_ERR_QMI_SEGMENT_ORDER
qmerrno.h, 929
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG
qmerrno.h, 929
eQCWWAN_ERR_QMI_SESSION_INACTIVE
qmerrno.h, 928
eQCWWAN_ERR_QMI_SESSION_INVALID
qmerrno.h, 928
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP
qmerrno.h, 928
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND
qmerrno.h, 929
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED
qmerrno.h, 928
eQCWWAN_ERR_QMI_SMSC_ADDR
qmerrno.h, 929
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE

qmerrno.h, 929
eQCWWAN_ERR_QMI_TPDU_TYPE
qmerrno.h, 929
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION
qmerrno.h, 927
eQCWWAN_ERR_QMI_UNKNOWN
qmerrno.h, 928
eQCWWAN_ERR_QMI_WIDTH
qmerrno.h, 930
eQCWWAN_ERR_RESET
qmerrno.h, 927
eQCWWAN_ERR_SWICM_AM_VERS_ERROR
qmerrno.h, 929
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS
qmerrno.h, 929
eQCWWAN_ERR_SWICM_END
qmerrno.h, 930
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS
qmerrno.h, 929
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID
qmerrno.h, 929
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID
qmerrno.h, 929
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID
qmerrno.h, 929
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED
qmerrno.h, 929
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED
qmerrno.h, 929
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED
qmerrno.h, 929
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS
qmerrno.h, 929
eQCWWAN_ERR_SWICM_SOCKET_IN_USE
qmerrno.h, 929
eQCWWAN_ERR_SWICM_START
qmerrno.h, 929
eQCWWAN_ERR_SWICM_TIMEOUT
qmerrno.h, 929
eQCWWAN_ERR_SWICM_V4DWN_V6DWN
qmerrno.h, 929
eQCWWAN_ERR_SWICM_V4DWN_V6UP
qmerrno.h, 929
eQCWWAN_ERR_SWICM_V4UP_V6DWN
qmerrno.h, 929
eQCWWAN_ERR_SWICM_V4UP_V6UP
qmerrno.h, 929
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED
qmerrno.h, 930
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND
qmerrno.h, 930
eQCWWAN_ERR_SWIDCS_END
qmerrno.h, 930
eQCWWAN_ERR_SWIDCS_FILEIO_ERR
qmerrno.h, 930
eQCWWAN_ERR_SWIDCS_IOCTL_ERR
qmerrno.h, 930
eQCWWAN_ERR_SWIDCS_START
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_END
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_INVALID_PATH
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_OPENING_DIR
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_OPENING_FILE
qmerrno.h, 930
eQCWWAN_ERR_SWIIM_START
qmerrno.h, 930
eQCWWAN_ERR_SWISM_END
qmerrno.h, 930
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND
qmerrno.h, 930
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED
qmerrno.h, 930
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG
qmerrno.h, 930

eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED
qmerrno.h, 930
eQCWWAN_ERR_SWISMS_START
qmerrno.h, 930
eQMI_LOC_SESS_STATUS_FAILURE
loc.h, 808
eQMI_LOC_SESS_STATUS_IN_PROGRESS
loc.h, 808
eQMI_LOC_SESS_STATUS_SUCCESS
loc.h, 808
eQMI_LOC_SESS_STATUS_TIMEOUT
loc.h, 808
eQOS
common.h, 737
eREQ
common.h, 738
eRMS
common.h, 737
eRSP
common.h, 738
eSAR
common.h, 737
eSDP_FWDWL_ERR_END
lite-fw.h, 799
eSDP_FWDWL_ERR_FAIL
lite-fw.h, 799
eSDP_FWDWL_ERR_FW_UPGRADE
lite-fw.h, 799
eSDP_FWDWL_ERR_FW_VERSION_FAIL
lite-fw.h, 799
eSDP_FWDWL_ERR_GENERAL
lite-fw.h, 798
eSDP_FWDWL_ERR_INVALID_DEV
lite-fw.h, 799
eSDP_FWDWL_ERR_INVALID_PATH
lite-fw.h, 799
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED
lite-fw.h, 799
eSDP_FWDWL_ERR_PATH_TOO_LONG
lite-fw.h, 799
eSDP_FWDWL_ERR_PRI_FAIL
lite-fw.h, 799
eSDP_FWDWL_ERR_SDK
lite-fw.h, 799
eSDP_FWDWL_ERR_SET_CBK
lite-fw.h, 799
eSDP_FWDWL_ERR_TIMEOUT
lite-fw.h, 799
eSDP_FWDWL_SUCCESS
lite-fw.h, 798
eSMS
common.h, 737
eSWILOC
common.h, 737
eSWIM2MCMD
common.h, 737
eSWIOMA
common.h, 737
eSWIOMAEXT
common.h, 737
eTIMEOUT_10_S
common.h, 738
eTIMEOUT_20_S
common.h, 738
eTIMEOUT_2_S
common.h, 738
eTIMEOUT_300_S
common.h, 738
eTIMEOUT_30_S
common.h, 738
eTIMEOUT_5_S
common.h, 738
eTIMEOUT_60_S
common.h, 738
eTIMEOUT_8_S
common.h, 738
eTIMEOUT_DEFAULT
common.h, 738
eTMD
common.h, 737
eUIM
common.h, 737
eVOICE
common.h, 737
eWDS
common.h, 737
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_END
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL

qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_FAIL
qmerrno.h, 931
eWDS_ERR_PROFILE_REG_RESULT_LIST_END
qmerrno.h, 931
ECIOThresListLen
nas_ECIOThresh, 133
eLOG_LEVEL
common.h, 737
EMTlv
NASQmiCbkNasSystemSelPrefInd, 235
eQCWWANError
qmerrno.h, 926
eQMI_SVC
common.h, 737
ERI_DATA_MAX_SIZE
dms.h, 747
ESNString
unpack_dms_GetDeviceSerialNumbers_t, 449
ETWSPLMNInfo
eTWSPLMNInfoTlv, 61
eTWSPLMNInfoTlv, 61
ETWSPLMNInfo, 61
TlvPresent, 61
ETWSPLMNTlv
unpack_sms_SetNewSMSCallback_ind_t, 594
ETWSTlv
unpack_sms_SetNewSMSCallback_ind_t, 594
eTimeout
common.h, 737
eValid
LibPackTFTIDParams, 93
earfcn
nas_infoInterFreq, 148
nas_LTEInfoIntrafreq, 156
nas_ltePCI, 158
nas_umtsLTENbrCell, 216
earfcn0
nas_lteEARFCN, 150
earfcn1
nas_lteEARFCN, 150
ecio
cdmaSSInfo, 52
hdrSSInfo, 66
nas_ecioListElement, 133
nas_rxInfo, 195
nas_UMTSInfo, 215
tdscdmaSigInfoExt, 411
ecioDelta
nas_SLQSSignalStrengthsIndReq, 203
ecioInfo
nas_SLQSSignalStrengthsInformation, 204
ecioList
unpack_nas_SLQSGetSignalStrength_t, 531
ecioListLen
unpack_nas_SLQSGetSignalStrength_t, 531
ecioThresholdList
nas_SLQSSignalStrengthsIndReq, 203
ecioThresholdListLen
nas_SLQSSignalStrengthsIndReq, 203
egprsSupp
nas_GSMSysInfo, 142
egprsSuppValid
nas_GSMSysInfo, 142
elevation
loc_satelliteInfo, 106
EmerMode
NASEmergencyModeTlv, 225
emmConnState
nas_LTEInfo, 153
emmState
nas_LTEInfo, 153
emmSubState
nas_LTEInfo, 153
enable
pack_pds_SetPDSState_t, 300
pack_qos_SLQSSetQosEventCallback_t, 310
enabled
pack_rms_SetSMSWake_t, 310
unpack_rms_GetSMSWake_t, 590
unpack_wds_GetMobileIPProfile_t, 652
EnabledStatus
unpack_pds_SetPdsState_Ind_t, 564
EncryptProt
nas_protocolSubtypeElement, 186
EncryptedPIN1
pack_uim_ChangePin_t, 329
pack_uim_SetPinProtection_t, 331
pack_uim_UnblockPin_t, 339
EndProactiveSession
cat_EndProactiveSessionTlv, 50
EngineState
unpack_pds_SLQSGetGPSStateInfo_t, 568
engineState
unpack_loc_EngineState_Ind_t, 493

eqmiCbkSetStatus
sms.h, 948
eriData
eriDataparams, 61
eriDataLen
eriDataparams, 61
eriDataparams, 60
eriData, 61
eriDataLen, 61
eriFile
unpack_dms_SLQSGetERIFile_t, 471
error
unpack_wds_GetLastMobileIPError_t, 650
errorMask
unpack_cat_SetCATEventCallback_t, 438
errorRate
nas_errorRateListElement, 134
errorRateInfo
nas_SLQSSignalStrengthsInformation, 204
errorRateList
unpack_nas_SLQSGetSignalStrength_t, 531
errorRateListLen
unpack_nas_SLQSGetSignalStrength_t, 531
errorState
slotInf, 392
uim_slotInfo, 435
esn
unpack_dms_GetSerialNumbers_t, 459
esnSize
unpack_dms_GetDeviceSerialNumbers_t, 449
EspSpi
unpack_qos_swiQosFilter_t, 582
EtwsMessageInfo
sMSEtwsMessageTlv, 407
event
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 579
unpack_qos_SLQSSetQosStatusCallback_ind_t, 580
event_Index
unpack_cat_SetCatEventCallback_ind_t, 438
EventID
cat_commonEventTlv, 49
EventLength
cat_commonEventTlv, 49
eventMask
pack_cat_SetCATEventCallback_t, 240
pack_uim_SLQSUIMEventRegister_t, 333
unpack_uim_SLQSUIMEventRegister_t, 615
eventRegister
pack_loc_EventRegister_t, 260
eventType
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 605
evrcCapability
voice_prefVoiceSO, 709
executingImage
FMSImageIDEntries, 64
exponent
unpack_qos_pktErrRate_t, 572
extBit
voice_calledPartySubAdd, 688
ExtErrorCode
PackCreateProfileOut, 384
extPowerState
pack_loc_SetExtPowerState_t, 260
extendedErrorCode
unpack_wds_SLQSDeleteProfile_t, 658

FLOAT
SwiDataTypes.h, 984
FMSImageElement, 61
buildId, 62
buildIdLength, 62
imageId, 62
imageType, 62
FMSImageIDEntries, 63
executingImage, 64
imageIDElement, 64
imageIDSize, 64
imageType, 64
maxImages, 64
FMSImageIdElement, 62
buildID, 63
buildIDLength, 63
failureCount, 63
imageID, 63
storageIndex, 63
FMSImageList, 64
imageIDEntries, 64
listSize, 64
FMSPrefImageList, 65
listEntries, 65
listSize, 65
FOTAUpdate
pack_swioma_SLQSOMADMSetSettings_t, 328
unpack_swioma_SLQSOMADMGetSettings_t, 609
FOTAdownload
pack_swioma_SLQSOMADMSetSettings_t, 328
unpack_swioma_SLQSOMADMGetSettings_t, 609
facility
pack_dms_UIMGetControlKeyStatus_t, 251
pack_dms_UIMSetControlKeyProtection_t, 253
pack_dms_UIMUnblockControlKey_t, 254
facilityCk
pack_dms_UIMSetControlKeyProtection_t, 253
pack_dms_UIMUnblockControlKey_t, 254
facilityState
pack_dms_UIMSetControlKeyProtection_t, 253
unpack_dms_UIMGetControlKeyStatus_t, 480
failureCount
FMSImageIdElement, 63
family
pack_wds_GetDefaultProfileNum_t, 360
pack_wds_SetDefaultProfileNum_t, 366
feature
uim_depersonalizationInformation, 421

uim_personalizationStatus, 427
fileID
uim_fileAttributes, 425
uim_fileInfo, 426
fileIndex
pack_uim_ReadTransparent_t, 330
pack_uim_SLQSUIMGetFileAttributes_t, 334
fileSize
uim_fileAttributes, 425
fileType
uim_fileAttributes, 425
fill_pack_ctx
common.h, 738
fill_sdu_hdr
common.h, 738
filterId
LibPackTFTIDParams, 93
fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 325
unpack_swiloc_SwiLocGetAutoStart_t, 604
fix_rate_reported
unpack_swiloc_SwiLocGetAutoStart_t, 604
fix_type
pack_swiloc_SwiLocSetAutoStart_t, 325
unpack_swiloc_SwiLocGetAutoStart_t, 604
fix_type_reported
unpack_swiloc_SwiLocGetAutoStart_t, 604
fixAccuracy
pack_pds_StartPDSTrackingSessionExt_t, 308
fixCount
pack_pds_StartPDSTrackingSessionExt_t, 308
fixInterval
pack_pds_StartPDSTrackingSessionExt_t, 308
fixTimeout
pack_pds_StartPDSTrackingSessionExt_t, 308
flags
sensorData_t, 389
flowLabel
LibPackTFTIDParams, 93
fms.h, 792
GetValidFwPriCombinations, 793
pack_fms_GetImagesPreference, 793
pack_fms_GetStoredImages, 794
pack_fms_SetImagesPreference, 794
unpack_fms_GetImagesPreference, 794
unpack_fms_GetStoredImages, 794
unpack_fms_SetImagesPreference, 794
Forbidden
nas_QmiNas3GppNetworkInfo, 188
forceFlag
pack_pds_SLQSPDSInjectAbsoluteTimeReference-_t, 303
ForceRev0
unpack_nas_GetCDMANetworkParameters_t, 508
format
sMSTransferRouteMTMessage, 410
ForwardMac
nas_protocolSubtypeElement, 186
fqdnAddr
wds_PCSCFFQDNAddress, 723
fqdnLen
wds_PCSCFFQDNAddress, 723
freeSlots
sms_maxStorageSizeResp, 398
freq
nas_PhyCaAggPcellInfo, 177
nas_PhyCaAggScellIndType, 179
nas_PhyCaAggScellInfo, 182
NASPhyCaAggPcellInfo, 230
NASPhyCaAggScellIndType, 232
NASPhyCaAggScellInfo, 233
freqsLen
nas_LTEInfoInterfreq, 154
nas_LTEInfoNeighboringGSM, 157
nas_LTEInfoNeighboringWCDMA, 158
fromServiceId
sms_BroadcastConfig, 393
function
pack_swiloc_SwiLocSetAutoStart_t, 325
unpack_swiloc_SwiLocGetAutoStart_t, 604
function_reported
unpack_swiloc_SwiLocGetAutoStart_t, 604
FwAutoCheck
unpack_swioma_SLQSOMADMGetSettings_t, 609
FwAutoSDM
unpack_swiavms_SLQSAVMSGetSettings_t, 601
FwAutoUpdate
pack_swiavms_SLQSAVMSSetSettings_t, 322
unpack_swiavms_SLQSAVMSGetSettings_t, 601
FwAutodownload
pack_swiavms_SLQSAVMSSetSettings_t, 322
unpack_swiavms_SLQSAVMSGetSettings_t, 601
FwAvailability
unpack_swioma_SLQSOMADMStartSession_t, 610
fwdloadsize
unpack_omaDmFotaTlv_t, 557
fwloadComplete
unpack_omaDmFotaTlv_t, 557
fwvers
unpack_dms_SLQSSwiGetFirmwareCurr_t, 474

gDlBitRate
LibPackQosClassID, 91
GPRSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 667
GPSLPM
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
GPSSel
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
GSMRSSIThreshListLen
nas_GSMRSSIThresh, 138
GSMSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 542
gUlBitRate

LibPackQosClassID, 91
GWAOPTlv
NASQmiCbkNasSystemSelPrefInd, 235
GWAcqOrderPref
NASGWAcqOrderPrefTlv, 226
GWAddressV4
unpack_wds_SLQSGetRuntimeSettings_t, 667
gcdumpString
crashInformation, 54
gcdumpStrlen
crashInformation, 54
geoSysIdx
nas_AddCDMASysInfo, 112
nas_AddSysInfo, 113
geranArfcn
nas_geranInstInfo, 136
geranBsicBcc
nas_geranInstInfo, 136
geranBsicNcc
nas_geranInstInfo, 137
geranInst
nas_UMTSInfo, 215
GeranInstInfo
nas_UMTSInfo, 215
geranRssi
nas_geranInstInfo, 137
get_version
common.h, 738
getAllCallInfo
voice_arrCallInfo, 682
GetAllCallRmtPtyName
voice_arrRemotePartyName, 685
GetCustomFeatureV2
unpack_dms_GetCustFeaturesV2_t, 444
GetValidFwPriCombinations
fms.h, 793
glo_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 568
glo_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 568
glo_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 568
glo_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 568
globalCellId
nas_LTEInfoIntrafreq, 156
glog
common.h, 739
gloglvl
common.h, 739
gnssSvId
loc_satelliteInfo, 106
gnssSvUsedList
loc_svUsedforFix, 109
gnssSvUsedList_len
loc_svUsedforFix, 109
Gpp2TimeZone
unpack_nas_SLQSGetServingSystem_t, 528
GppNetworkDSTAdjustment
unpack_nas_SLQSGetServingSystem_t, 528
GppTimeZone
unpack_nas_SLQSGetServingSystem_t, 528
gps_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 568
gps_ephemeris_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 568
gps_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 569
gps_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 569
GpsEnable
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
gpsTimeOfWeekMs
loc_gpsTime, 101
gpsWeek
loc_gpsTime, 101
grntDownlinkBitrate
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
grntUplinkBitrate
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
gsmAmrStat
voice_curAMRConfig, 701
GsmCellInfo
nas_lteGsmCellInfo, 151
gsmUmtsDl
pack_nas_SLQSNasSwiIndicationRegister_t, 288
gsmUmtsUl
pack_nas_SLQSNasSwiIndicationRegister_t, 288
guaranteedRate
unpack_qos_dataRate_t, 570
gwAddressV6
wds_IPV6GWAddressInfo, 723
gwV6PrefixLen
wds_IPV6GWAddressInfo, 723
gyroData
pack_loc_SLQSLOCInjectSensorData_t, 268
gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 268
gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 269

HASPI
unpack_wds_GetMobileIPProfile_t, 652
HAState
unpack_wds_GetMobileIPProfile_t, 653
HDOP
loc_precisionDilution, 104
HDRECIOThreshListLen
nas_HDRECIOThresh, 142
HDRIOThreshListLen
nas_HDRIOThresh, 143
HDRRSSIThreshListLen
nas_HDRRSSIThresh, 143
HDRSINRThresListLen

nas_HDRSINRThresh, 144
HDRSINRThreshListLen
nas_HDRSINRThreshold, 145
HDRSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 542
HWVersion
nas_DeviceConfigDetail, 131
HardwareControlledMode
unpack_dms_GetPower_t, 457
has_PositionDataNMEA
unpack_pds_SetEventReport_Ind_t, 564
has_SessionStatus
unpack_pds_SetEventReport_Ind_t, 564
has_accelTemp
pack_loc_SLQSLOCInjectSensorData_t, 269
has_acceleroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 269
has_accleroData
pack_loc_SLQSLOCInjectSensorData_t, 269
has_altitudeSrcInfo
pack_loc_SLQSLOCInjectPosition_t, 266
has_altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 266
has_altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 266
has_confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 270
has_dLatitude
unpack_pds_SetEventReport_Ind_t, 563
has_dLongitude
unpack_pds_SetEventReport_Ind_t, 563
has_gyroData
pack_loc_SLQSLOCInjectSensorData_t, 269
has_gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 269
has_gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 269
has_horConfidence
pack_loc_SLQSLOCInjectPosition_t, 266
has_horReliability
pack_loc_SLQSLOCInjectPosition_t, 266
has_horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 266
has_latitude
pack_loc_SLQSLOCInjectPosition_t, 266
has_longitude
pack_loc_SLQSLOCInjectPosition_t, 266
has_opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 269
has_posSrc
unpack_pds_SetEventReport_Ind_t, 564
has_positionSrc
pack_loc_SLQSLOCInjectPosition_t, 266
has_rawHorConfidence
pack_loc_SLQSLOCInjectPosition_t, 266
has_rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 266
has_serverAddrTypeMask
pack_loc_SLQSLOCGetServer_t, 262
has_timestampAge
pack_loc_SLQSLOCInjectPosition_t, 266
has_timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 266
has_vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 266
has_vertRelicability
pack_loc_SLQSLOCInjectPosition_t, 266
has_vertUnc
pack_loc_SLQSLOCInjectPosition_t, 267
hdrActiveProt
nas_HDRSysInfo, 147
hdrActiveProtValid
nas_HDRSysInfo, 147
hdrHybrid
nas_detailSvcInfo, 130
HdrPersonality
unpack_nas_SLQSGetServingSystem_t, 528
hdrPersonality
nas_HDRSysInfo, 147
NASServingSystemInfo, 237
hdrPersonalityValid
nas_HDRSysInfo, 147
hdrSSInfo, 65
ecio, 66
io, 66
rssi, 66
sinr, 66
hdrSrvStatus
nas_detailSvcInfo, 130
healthStatus
loc_satelliteInfo, 106
helper_get_resp_ctx
common.h, 738
helper_get_xid
common.h, 738
helper_set_log_func
common.h, 739
helper_set_log_lvl
common.h, 739
homeOrigVoiceSO
voice_prefVoiceSO, 709
homePageVoiceSO
voice_prefVoiceSO, 709
horConfidence
pack_loc_SLQSLOCInjectPosition_t, 267
horReliability
pack_loc_SLQSLOCInjectPosition_t, 267
horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 267
HorizontalUncertainty
unpack_pds_SLQSGetGPSStateInfo_t, 569
hotSwap
uim_hotSwapStatus, 426
hotSwapLength
uim_hotSwapStatus, 426
hour

nas_timeInfo, 212
nas_UniversalTime, 218
hsCallStatus
nas_WCDMASysInfo, 223
hsCallStatusValid
nas_WCDMASysInfo, 223
hsInd
nas_WCDMASysInfo, 224
hsIndValid
nas_WCDMASysInfo, 224
hwType
wds_DHCPv4HWConfig, 718
wdsDhcpv4HwConfig, 729
hwVer
unpack_dms_GetHardwareRevision_t, 453

iLTEbandValue
nas_PhyCaAggPcellInfo, 177
nas_PhyCaAggScellInfo, 182
NASPhyCaAggPcellInfo, 230
NASPhyCaAggScellInfo, 233
IMCNflag
unpack_wds_SLQSGetRuntimeSettings_t, 667
IMEIString
unpack_dms_GetDeviceSerialNumbers_t, 449
IMG_MASK_CLEAR
lite-fw.h, 797
IMG_MASK_GENERIC
lite-fw.h, 797
IMG_MASK_MDM
lite-fw.h, 797
IMG_MASK_PRI
lite-fw.h, 797
IMSInfo
sMSOnIMSTlv, 409
IMSTlv
unpack_sms_SetNewSMSCallback_ind_t, 594
INT32
SwiDataTypes.h, 984
INT8
SwiDataTypes.h, 984
IOThresListLen
nas_IOThresh, 149
IPAddress
nas_DataStatusDetail, 129
IPAddressV6
ipv6AddressInfo, 67
wds_IPV6AddressInfo, 722
IPFamSupport
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
IPFamilyPreference
pack_wds_SLQSSetIPFamilyPreference_t, 377
unpack_wds_SLQSGetRuntimeSettings_t, 667
IPSECSPI
LibPackTFTIDParams, 93
IPV6AddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 667
IPV6GWAddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 667
IPV6PrefixLen
ipv6AddressInfo, 67
wds_IPV6AddressInfo, 722
IPv4
unpack_wds_SLQSGetRuntimeSettings_t, 667
IPv4Addr
wds_IPv4AdTlv, 722
IPv4AddrTlv
unpack_wds_DHCPv4ClientLease_ind_t, 645
IPv4DstAddr
unpack_qos_swiQosFilter_t, 583
IPv4SrcAddr
unpack_qos_swiQosFilter_t, 583
IPv4Tos
unpack_qos_swiQosFilter_t, 583
IPv6DstAddr
unpack_qos_swiQosFilter_t, 583
IPv6Label
unpack_qos_swiQosFilter_t, 583
IPv6SrcAddr
unpack_qos_swiQosFilter_t, 583
IPv6TrafCls
unpack_qos_swiQosFilter_t, 583
Id
unpack_qos_swiQosFilter_t, 582
id
loc_BdsSV, 97
loc_SV, 108
nas_CSGID, 125
pack_dms_UIMChangePIN_t, 251
pack_dms_UIMSetPINProtection_t, 253
pack_dms_UIMUnblockPIN_t, 254
pack_dms_UIMVerifyPIN_t, 255
unpack_qos_QosFlowInfoState_t, 574
unpack_qos_SLQSSetQosStatusCallback_ind_t, 580
id_length
DMScustSettingInfo, 58
IdleState
nas_protocolSubtypeElement, 186
image_info_t, 66
buildID, 67
buildIDLen, 67
imageType, 67
uniqueID, 67
imageID
FMSImageIdElement, 63
imageIDElement
FMSImageIDEntries, 64
imageIDEntries
FMSImageList, 64
imageIDSize
FMSImageIDEntries, 64
imageId
FMSImageElement, 62
imageList
unpack_fms_GetStoredImages_t, 485

ImageListSize
unpack_fms_GetImagesPreference_t, 485
imageListSize
pack_fms_SetImagesPreference_t, 257
imageType
FMSImageElement, 62
FMSImageIDEntries, 64
image_info_t, 67
ImageTypes
unpack_fms_SetImagesPreference_t, 486
ImageTypesSize
unpack_fms_SetImagesPreference_t, 486
imagelistSize
unpack_fms_GetStoredImages_t, 485
imei_no
unpack_dms_GetSerialNumbers_t, 459
imeiSize
unpack_dms_GetDeviceSerialNumbers_t, 449
imeiSvnSize
unpack_dms_GetDeviceSerialNumbers_t, 449
ImeiSvnString
unpack_dms_GetDeviceSerialNumbers_t, 449
imeisv_svn
unpack_dms_GetSerialNumbers_t, 459
imgType
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 475
imsRegState
nas_CommInfo, 124
imsi
unpack_dms_GetIMSI_t, 454
imsi_11_12
nas_CDMASysInfoExt, 122
imsiM1112
nas_minBasedIMSI, 168
imsiMS1
nas_minBasedIMSI, 168
imsiMS2
nas_minBasedIMSI, 168
imsiT1112
nas_trueIMSI, 213
imsiTS1
nas_trueIMSI, 213
imsiTS2
nas_trueIMSI, 213
imsiTaddrNum
nas_trueIMSI, 213
InUse
nas_QmiNas3GppNetworkInfo, 188
includes_pcs_digit
nas_QmisNasPcsDigit, 190
index
pack_wds_GetMobileIPProfile_t, 361
pack_wds_SetDefaultProfileNum_t, 366
pack_wds_SetMobileIPProfile_t, 369
unpack_qos_swiQosFilter_t, 583
unpack_qos_swiQosFlow_t, 587
unpack_wds_GetDefaultProfileNum_t, 650
index1xPri
uim_cardStatus, 419
index1xSec
uim_cardStatus, 419
indexGwPri
uim_cardStatus, 419
indexGwSec
uim_cardStatus, 419
Info
unpack_nas_SLQSNasSwiOTAMessageCallback-_ind_t, 546
unpack_nas_SLQSSetSysSelectionPrefCallBack-_ind_t, 547
InfoInterfreq
nas_LTEInfoInterfreq, 154
insNmrCellInfo
nas_GERANInfo, 136
instancesSize
unpack_nas_GetRFInfo_t, 513
interval
pack_pds_SetPDSDefaults_t, 300
pack_pds_SetXTRAAutomaticDownload_t, 302
pack_wds_SLQSSetWdsEventCallback_t, 378
Io
unpack_nas_SLQSGetSignalStrength_t, 531
io
hdrSSInfo, 66
nas_SLQSSignalStrengthsInformation, 204
ioDelta
nas_SLQSSignalStrengthsIndReq, 203
Iono_valid
unpack_pds_SLQSGetGPSStateInfo_t, 569
ipAddress
pack_wds_SetDefaultProfile_t, 365
ipFamily
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 669
ipVersion
LibPackTFTIDParams, 93
ipaddr
unpack_wds_GetDefaultProfile_t, 649
ipaddrv6
unpack_wds_GetDefaultProfile_t, 649
ipv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 674
ipv4GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 674
ipv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 675
ipv6AddressInfo, 67
IPAddressV6, 67
IPV6PrefixLen, 67
ipv6GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 675
is856SysId

nas_HDRSysInfo, 147
is856SysIdValid
nas_HDRSysInfo, 147
is_DataRate_Available
unpack_qos_swiQosFlow_t, 587
is_EspSpi_Available
unpack_qos_swiQosFilter_t, 583
is_IPv4DstAddr_Available
unpack_qos_swiQosFilter_t, 583
is_IPv4SrcAddr_Available
unpack_qos_swiQosFilter_t, 583
is_IPv4Tos_Available
unpack_qos_swiQosFilter_t, 583
is_IPv6DstAddr_Available
unpack_qos_swiQosFilter_t, 583
is_IPv6Label_Available
unpack_qos_swiQosFilter_t, 583
is_IPv6SrcAddr_Available
unpack_qos_swiQosFilter_t, 583
is_IPv6TrafCls_Available
unpack_qos_swiQosFilter_t, 583
is_Id_Available
unpack_qos_swiQosFilter_t, 583
is_Jitter_Available
unpack_qos_swiQosFlow_t, 587
is_Latency_Available
unpack_qos_swiQosFlow_t, 587
is_LteBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 471
is_LteQci_Available
unpack_qos_swiQosFlow_t, 587
is_MaxAllowedPktSz_Available
unpack_qos_swiQosFlow_t, 587
is_MinPolicedPktSz_Available
unpack_qos_swiQosFlow_t, 587
is_NxtHdrProto_Available
unpack_qos_swiQosFilter_t, 583
is_PktErrRate_Available
unpack_qos_swiQosFlow_t, 587
is_Precedence_Available
unpack_qos_swiQosFilter_t, 583
is_ProfileId3GPP2_Available
unpack_qos_swiQosFlow_t, 587
is_RxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 574
is_TCPDstPort_Available
unpack_qos_swiQosFilter_t, 583
is_TCPSrcPort_Available
unpack_qos_swiQosFilter_t, 583
is_TdsBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 471
is_TokenBucket_Available
unpack_qos_swiQosFlow_t, 587
is_TrafficClass_Available
unpack_qos_swiQosFlow_t, 587
is_TranDstPort_Available
unpack_qos_swiQosFilter_t, 583
is_TranSrcPort_Available
unpack_qos_swiQosFilter_t, 583
is_TxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 574
is_UDPDstPort_Available
unpack_qos_swiQosFilter_t, 583
is_UDPSrcPort_Available
unpack_qos_swiQosFilter_t, 583
is_val_3GPP2Pri_Available
unpack_qos_swiQosFlow_t, 587
is_val_3GPPImCn_Available
unpack_qos_swiQosFlow_t, 587
is_val_3GPPResResidualBER_Available
unpack_qos_swiQosFlow_t, 587
is_val_3GPPSigInd_Available
unpack_qos_swiQosFlow_t, 587
is_val_3GPPTraHdlPri_Available
unpack_qos_swiQosFlow_t, 587
isEmpty
voice_getAllCallInformation, 703
isInTraffic
nas_txInfo, 213
isNewFlow
unpack_qos_QosFlowInfoState_t, 574
isPrefDataPath
nas_GSMSrvStatusInfo, 139
nas_SrvStatusInfo, 205
isRadioTuned
nas_rxInfo, 195
nas_RxSigInfo, 196
isSysForbidden
nas_detailSvcInfo, 130
nas_sysInfoCommon, 207
isSysForbiddenValid
nas_sysInfoCommon, 207
isSysPrlMatch
nas_CDMASysInfo, 121
nas_HDRSysInfo, 147
isSysPrlMatchValid
nas_CDMASysInfo, 121
nas_HDRSysInfo, 147
IsVoiceEnabled
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443

Jitter
unpack_qos_swiQosFlow_t, 587

KeyExchange
nas_protocolSubtypeElement, 186

LITE_VOICE_SUPS_SRV_CLASS_DATA
voice.h, 1017
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITAS-YNC
voice.h, 1018
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSY-NC
voice.h, 1018
LITE_VOICE_SUPS_SRV_CLASS_FAX

voice.h, 1018
LITE_VOICE_SUPS_SRV_CLASS_NONE
voice.h, 1017
LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS
voice.h, 1018
LITE_VOICE_SUPS_SRV_CLASS_PADACCESS
voice.h, 1018
LITE_VOICE_SUPS_SRV_CLASS_SMS
voice.h, 1018
LITE_VOICE_SUPS_SRV_CLASS_VOICE
voice.h, 1017
LITEQMI_QMI_CBK_PARAM_NOCHANGE
sms.h, 948
LITEQMI_QMI_CBK_PARAM_RESET
sms.h, 948
LITEQMI_QMI_CBK_PARAM_SET
sms.h, 948
LBPTlv
NASQmiCbkNasSystemSelPrefInd, 235
LOCEVENTMASKGNSSSVINFO
loc.h, 806
LOCEVENTMASKNMEA
loc.h, 807
LOCEVENTMASKWIFIREQ
loc.h, 808
LPCSTR
SwiDataTypes.h, 984
LTEAttachProfile
unpack_wds_SLQSGet3GPPConfigItem_t, 659
LTEAttachProfileList
unpack_wds_SLQSGet3GPPConfigItem_t, 660
LTEAttachProfileListLen
pack_wds_SLQSSet3GPPConfigItem_t, 376
unpack_wds_SLQSGet3GPPConfigItem_t, 660
LTEBandPref
NASLTEBandPreferenceTlv, 227
LTECphyCaInfo
unpack_nas_SlqsGetLTECphyCAInfo_t, 523
LTERSRPThreshListLen
nas_LTERSRPThresh, 159
LTERSRQThreshListLen
nas_LTERSRQThresh, 160
LTERSSIThreshListLen
nas_LTERSSIThresh, 160
LTESNRThresListLen
nas_LTESNRThresh, 163
LTESNRThreshListLen
nas_LTESNRThreshold, 164
LTESSInfo
unpack_nas_SLQSNasGetSigInfo_t, 542
LTEWCDMACellInfo
nas_LTEInfoNeighboringWCDMA, 158
Lac
unpack_nas_SLQSGetServingSystem_t, 528
lac
nas_GERANInfo, 136
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_UMTSInfo, 215
nas_WCDMASysInfo, 224
lac1
nas_OperatorPLMNData, 176
lac2
nas_OperatorPLMNData, 176
lacValid
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_WCDMASysInfo, 224
language
sms_CDMABroadcastConfig, 394
lastCallDataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 664
lastCallDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 662
lastCallRXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 664
lastCallTXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 664
LastErrCode
nas_DataStatusDetail, 129
Latency
unpack_qos_swiQosFlow_t, 587
Latitude
unpack_pds_SLQSGetGPSStateInfo_t, 569
latitude
pack_loc_SLQSLOCInjectPosition_t, 267
leapSeconds
nas_qaQmi3Gpp2TimeZone, 187
leaseState
wds_DHCPLeaseStateTlv, 716
len
loc_BdsSVInfo, 97
loc_SVInfo, 108
unpack_nas_GetSignalStrengths_t, 516
length
sMSCAddress, 406
sMSEtwsMessage, 407
sMSTransferRouteMTMessage, 410
uim_readTransparentInfo, 428
LibPackGPRSRequestedQoS, 67
delayClass, 68
meanThroughputClass, 68
peakThroughputClass, 68
precedenceClass, 68
reliabilityClass, 68
LibPackQosClassID, 90
gDlBitRate, 91
gUlBitRate, 91
maxDlBitRate, 91
maxUlBitRate, 91
QCI, 91
LibPackTFTIDParams, 91
destPortRangeEnd, 92
destPortRangeStart, 93
eValid, 93

filterId, 93
flowLabel, 93
IPSECSPI, 93
ipVersion, 93
nextHeader, 93
pSourceIP, 93
sourceIPMask, 93
srcPortRangeEnd, 93
srcPortRangeStart, 93
tosMask, 93
LibPackUMTSQoS, 93
deliveryErrSDU, 95
grntDownlinkBitrate, 95
grntUplinkBitrate, 95
maxDownlinkBitrate, 95
maxSDUSize, 95
maxUplinkBitrate, 95
qosDeliveryOrder, 95
resBerRatio, 95
sduErrorRatio, 95
trafficClass, 95
trafficPriority, 95
transferDelay, 95
LibPackUMTSReqQoSSigInd, 95
SigInd, 96
UMTSReqQoS, 96
LibPackprofile_3GPP, 79
pAPNClass, 84
pAPNDisabledFlag, 84
pAPNName, 84
pAPNnameSize, 84
pAddrAllocPref, 84
pAuthenticationPref, 84
pGPRSMinimumQoS, 84
pGPRSRequestedQos, 84
pIPv4AddrPref, 84
pIPv6AddPref, 84
pImCnFlag, 84
pPDNInactivTimeout, 84
pPDPtype, 84
pPassword, 84
pPasswordSize, 84
pPcscfAddrUsingDhcp, 84
pPcscfAddrUsingPCO, 84
pPdpAccessConFlag, 84
pPdpContext, 84
pPdpDataCompType, 84
pPdpHdrCompType, 84
pPriDNSIPv4AddPref, 84
pPriDNSIPv6addpref, 84
pPrimaryID, 84
pProfilename, 85
pProfilenameSize, 85
pQosClassID, 85
pSecDNSIPv4AddPref, 85
pSecDNSIPv6addpref, 85
pSecondaryFlag, 85
pTFTID1Params, 85
pTFTID2Params, 85
pUMTSMinQoS, 85
pUMTSMinQosSigInd, 85
pUMTSReqQoS, 85
pUMTSReqQoSSigInd, 85
pUsername, 85
pUsernameSize, 85
LibPackprofile_3GPP2, 85
pAPNClass3GPP2, 89
pAPNEnabled3GPP2, 89
pAllowLinger, 89
pApnString, 89
pApnStringSize, 89
pAppPriority, 89
pAppType, 89
pAuthPassword, 89
pAuthPassword_tSize, 89
pAuthProtocol, 89
pAuthRetryCount, 90
pAuthTimeout, 90
pDataMode, 90
pDataRate, 90
pIpcpAckTimeout, 90
pIpcpCreqRetryCount, 90
pIsPcscfAddressNedded, 90
pLcpAckTimeout, 90
pLcpCreqRetryCount, 90
pNegoDnsSrvrPref, 90
pPDNInactivTimeout3GPP2, 90
pPdnType, 90
pPppSessCloseTimer1x, 90
pPppSessCloseTimerDO, 90
pPriV6DnsAddress, 90
pPrimaryV4DnsAddress, 90
pRATType, 90
pSecV6DnsAddress, 90
pSecondaryV4DnsAddress, 90
pUserId, 90
pUserIdSize, 90
libSDP_BuildImagesPreferenceRequest
lite-fw.h, 797
libSDP_CalculateImageMask
lite-fw.h, 797
libSDP_CheckValidFirmwareInfo
lite-fw.h, 797
libSDP_DownloadFW
lite-fw.h, 797
libSDP_ExtractFirmwareParametersByPath
lite-fw.h, 797
libSDP_FirmwareInfo
lite-fw.h, 797
libSDP_GetModelFamily
lite-fw.h, 797
libSDP_GetVersion
lite-fw.h, 797
libSDP_getFileType
lite-fw.h, 797
LibpackProfile3GPP, 68

pAPNClass, 73
pAPNDisabledFlag, 73
pAPNName, 73
pAPNnameSize, 73
pAddrAllocPref, 73
pAuthenticationPref, 73
pGPRSMinimumQoS, 73
pGPRSRequestedQos, 73
pIPv4AddrPref, 73
pIPv6AddPref, 73
pImCnFlag, 73
pPDNInactivTimeout, 73
pPDPtype, 73
pPassword, 73
pPasswordSize, 73
pPcscfAddrUsingDhcp, 73
pPcscfAddrUsingPCO, 73
pPdpAccessConFlag, 73
pPdpContext, 73
pPdpDataCompType, 73
pPdpHdrCompType, 73
pPriDNSIPv4AddPref, 73
pPriDNSIPv6addpref, 73
pPrimaryID, 73
pProfilename, 74
pProfilenameSize, 74
pQosClassID, 74
pSecDNSIPv4AddPref, 74
pSecDNSIPv6addpref, 74
pSecondaryFlag, 74
pTFTID1Params, 74
pTFTID2Params, 74
pUMTSMinQoS, 74
pUMTSMinQosSigInd, 74
pUMTSReqQoS, 74
pUMTSReqQoSSigInd, 74
pUsername, 74
pUsernameSize, 74
LibpackProfile3GPP2, 74
pAPNClass3GPP2, 78
pAPNEnabled3GPP2, 78
pAllowLinger, 78
pApnString, 78
pApnStringSize, 78
pAppPriority, 78
pAppType, 78
pAuthPassword, 78
pAuthPasswordSize, 78
pAuthProtocol, 78
pAuthRetryCount, 79
pAuthTimeout, 79
pDataMode, 79
pDataRate, 79
pIpcpAckTimeout, 79
pIpcpCreqRetryCount, 79
pIsPcscfAddressNedded, 79
pLcpAckTimeout, 79
pLcpCreqRetryCount, 79
pNegoDnsSrvrPref, 79
pPDNInactivTimeout3GPP2, 79
pPdnType, 79
pPppSessCloseTimer1x, 79
pPppSessCloseTimerDO, 79
pPriV6DnsAddress, 79
pPrimaryV4DnsAddress, 79
pRATType, 79
pSecV6DnsAddress, 79
pSecondaryV4DnsAddress, 79
pUserId, 79
pUserIdSize, 79
libsdp_SetReadBlockSize
lite-fw.h, 797
libsdp_set_log_func
lite-fw.h, 797
libsdplogger
lite-fw.h, 797
lineValue
pack_voice_SLQSVoiceALSSelectLine_t, 342
linkage
altSrcInfo_t, 43
list_type
DMScustSettingList, 58
DMSgetCustomInput, 60
pack_dms_GetCustFeaturesV2_t, 241
listEntries
FMSPrefImageList, 65
listSize
FMSImageList, 64
FMSPrefImageList, 65
lite-fw.h
eFW_TYPE_CWE, 798
eFW_TYPE_CWE_NVU, 798
eFW_TYPE_INVALID, 798
eFW_TYPE_MBN, 798
eFW_TYPE_MBN_GOBI, 798
eFW_TYPE_NVU, 798
eFW_TYPE_SPK, 798
eModel_9X15, 799
eModel_9X30, 799
eModel_9x07, 799
eModel_Unknown, 799
eModel_WP9X15, 799
eSDP_FWDWL_ERR_END, 799
eSDP_FWDWL_ERR_FAIL, 799
eSDP_FWDWL_ERR_FW_UPGRADE, 799
eSDP_FWDWL_ERR_FW_VERSION_FAIL, 799
eSDP_FWDWL_ERR_GENERAL, 798
eSDP_FWDWL_ERR_INVALID_DEV, 799
eSDP_FWDWL_ERR_INVALID_PATH, 799
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED, 799
eSDP_FWDWL_ERR_PATH_TOO_LONG, 799
eSDP_FWDWL_ERR_PRI_FAIL, 799
eSDP_FWDWL_ERR_SDK, 799
eSDP_FWDWL_ERR_SET_CBK, 799
eSDP_FWDWL_ERR_TIMEOUT, 799

eSDP_FWDWL_SUCCESS, 798
lite-fw.h, 795
IMG_MASK_CLEAR, 797
IMG_MASK_GENERIC, 797
IMG_MASK_MDM, 797
IMG_MASK_PRI, 797
libSDP_BuildImagesPreferenceRequest, 797
libSDP_CalculateImageMask, 797
libSDP_CheckValidFirmwareInfo, 797
libSDP_DownloadFW, 797
libSDP_ExtractFirmwareParametersByPath, 797
libSDP_FirmwareInfo, 797
libSDP_GetModelFamily, 797
libSDP_GetVersion, 797
libSDP_getFileType, 797
libsdp_SetReadBlockSize, 797
libsdp_set_log_func, 797
libsdplogger, 797
litefw_BuildImagesPreferenceRequest, 799
litefw_CalculateImageMask, 799
litefw_CheckValidFirmwareInfo, 800
litefw_DownloadFW, 800
litefw_ExtractFirmwareParametersByPath, 801
litefw_FirmwareInfo, 797
litefw_Fw_Type, 798
litefw_GetModelFamily, 801
litefw_GetVersion, 802
litefw_Models, 799
litefw_SetReadBlockSize, 802
litefw_fwdwl_error_codes, 798
litefw_getFileType, 801
litefw_set_log_func, 802
litefw_switch_9x07_to_downloadmode, 802
litefwlogger, 798
liteQmiDataBearerMasks
wds.h, 1046
liteServiceClassInformation
voice.h, 1017
litefw_BuildImagesPreferenceRequest
lite-fw.h, 799
litefw_CalculateImageMask
lite-fw.h, 799
litefw_CheckValidFirmwareInfo
lite-fw.h, 800
litefw_DownloadFW
lite-fw.h, 800
litefw_ExtractFirmwareParametersByPath
lite-fw.h, 801
litefw_FirmwareInfo
lite-fw.h, 797
litefw_Fw_Type
lite-fw.h, 798
litefw_GetModelFamily
lite-fw.h, 801
litefw_GetVersion
lite-fw.h, 802
litefw_Models
lite-fw.h, 799
litefw_SetReadBlockSize
lite-fw.h, 802
litefw_fwdwl_error_codes
lite-fw.h, 798
litefw_getFileType
lite-fw.h, 801
litefw_set_log_func
lite-fw.h, 802
litefw_switch_9x07_to_downloadmode
lite-fw.h, 802
litefwlogger
lite-fw.h, 798
liteqmi_GetVersion
common.h, 739
liteqmi_helper_decode7bitAsciiEncString
common.h, 739
liteqmi_log
common.h, 739
loc.h
eQMI_LOC_SESS_STATUS_FAILURE, 808
eQMI_LOC_SESS_STATUS_IN_PROGRESS, 808
eQMI_LOC_SESS_STATUS_SUCCESS, 808
eQMI_LOC_SESS_STATUS_TIMEOUT, 808
loc.h, 803
LOCEVENTMASKNMEA, 807
LOCEVENTMASKWIFIREQ, 808
MAX_TEMP_DATA_LEN, 808
pack_loc_DeleteAssistData, 808
pack_loc_EventRegister, 808
pack_loc_SLQSLOCGetBestAvailPos, 810
pack_loc_SLQSLOCGetServer, 810
pack_loc_SLQSLOCInjectPosition, 810
pack_loc_SLQSLOCInjectSensorData, 811
pack_loc_SLQSLOCInjectUTCTime, 811
pack_loc_SLQSLOCSetCradleMountConfig, 811
pack_loc_SetExtPowerState, 809
pack_loc_SetOperationMode, 809
pack_loc_Start, 812
pack_loc_Stop, 812
unpack_loc_BestAvailPos_Ind, 812
unpack_loc_DeleteAssistData, 813
unpack_loc_DeleteAssistData_Ind, 813
unpack_loc_EngineState_Ind, 814
unpack_loc_EventNMEA_Ind, 814
unpack_loc_EventRegister, 814
unpack_loc_GetServer_Ind, 815
unpack_loc_GnssSvInfo_Ind, 815
unpack_loc_PositionRpt_Ind, 815
unpack_loc_SLQSLOCGetBestAvailPos, 817
unpack_loc_SLQSLOCGetServer, 817
unpack_loc_SLQSLOCInjectPosition, 818
unpack_loc_SLQSLOCInjectSensorData, 818
unpack_loc_SLQSLOCInjectUTCTime, 818
unpack_loc_SLQSLOCSetCradleMountConfig, 818
unpack_loc_SetExtPowerConfig_Ind, 816
unpack_loc_SetExtPowerState, 816

unpack_loc_SetOperationMode, 816
unpack_loc_SetOperationMode_Ind, 817
unpack_loc_Start, 819
unpack_loc_Stop, 819
loc_BdsSV, 96
id, 97
mask, 97
loc_BdsSVInfo, 97
len, 97
pSV, 97
loc_CellDb, 97
mask, 98
loc_ClkInfo, 98
mask, 99
loc_GnssData, 99
mask, 100
loc_IPv4Info, 101
address, 102
port, 102
TlvPresent, 102
loc_IPv6Info, 102
address, 102
port, 102
TlvPresent, 102
loc_LocApplicationInfo, 102
appNameLength, 103
appProviderLength, 103
appVersionLength, 103
appVersionValid, 103
pAppName, 103
pAppProvider, 103
pAppVersion, 103
loc_SV, 107
id, 108
mask, 108
system, 108
loc_SVInfo, 108
len, 108
pSV, 108
loc_gpsTime, 101
gpsTimeOfWeekMs, 101
gpsWeek, 101
loc_precisionDilution, 103
HDOP, 104
PDOP, 104
VDOP, 104
loc_satelliteInfo, 104
azimuth, 106
elevation, 106
gnssSvId, 106
healthStatus, 106
snr, 106
svInfoMask, 106
svListLen, 106
svStatus, 106
system, 106
validMask, 106
loc_sensorDataUsage, 106
aidingIndicatorMask, 107
usageMask, 107
loc_svUsedforFix, 108
gnssSvUsedList, 109
gnssSvUsedList_len, 109
loc_urlAddr, 109
address, 109
TlvPresent, 109
localTimeOffset
nas_qaQmi3Gpp2TimeZone, 187
logString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 475
logger
common.h, 737
longName
nas_PLMNNetworkNameData, 185
unpack_nas_SLQSGetPLMNName_t, 524
longNameCI
unpack_nas_SLQSGetPLMNName_t, 524
longNameEn
unpack_nas_SLQSGetPLMNName_t, 525
longNameLen
nas_PLMNNetworkNameData, 185
unpack_nas_SLQSGetPLMNName_t, 525
longNameSB
unpack_nas_SLQSGetPLMNName_t, 525
longNameSpareBits
nas_PLMNNetworkNameData, 185
Longitude
unpack_pds_SLQSGetGPSStateInfo_t, 569
longitude
pack_loc_SLQSLOCInjectPosition_t, 267
loopbackMode
pack_wds_SLQSSSetLoopback_t, 379
loopbackMultiplier
pack_wds_SLQSSSetLoopback_t, 379
LteBandCapability
unpack_dms_SLQSGetBandCapability_t, 471
lteEmmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 288
lteEmmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 288
lteEsmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 288
lteEsmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 288
LteGsmCellInfo
nas_LTEInfoNeighboringGSM, 157
LteQci
unpack_qos_swiQosFlow_t, 587
lteRsrpDelta
nas_SLQSSignalStrengthsIndReq, 203
lteRsrpinfo
nas_SLQSSignalStrengthsInformation, 204
lteSSInfo, 110
rsrp, 110
rsrq, 110
rssi, 110

snr, 110
lteSnrDelta
nas_SLQSSignalStrengthsIndReq, 203
lteSnrinfo
nas_SLQSSignalStrengthsInformation, 204
ltersrp
unpack_nas_SLQSGetSignalStrength_t, 531
ltesnr
unpack_nas_SLQSGetSignalStrength_t, 531

m_FwBuildId
CarrierImage_t, 47
m_FwImageId
CarrierImage_t, 47
m_PriBuildId
CarrierImage_t, 47
m_PriImageId
CarrierImage_t, 47
m_nCarrierId
CarrierImage_t, 47
m_nFolderId
CarrierImage_t, 47
m_nStorage
CarrierImage_t, 47
MACIndex
nas_NetworkStatEVDO, 173
MAX_BUILD_ID_LEN
dms.h, 747
MAX_ICCID_LENGTH
uim.h, 998
MAX_MSE_TWS_MSG
sms.h, 946
MAX_NO_OF_SLOTS
uim.h, 998
MAX_SENSOR_DATA_LEN
loc.h, 808
MAX_SLOTS_STATUS
uim.h, 998
MAX_SMS_LIST_SIZE
sms.h, 946
MAX_TEMP_DATA_LEN
loc.h, 808
MAXVOICEUSSDLENGTH
voice.h, 1017
MCC
nas_CDMASysInfo, 121
nas_CDMASysInfoExt, 122
nas_currentPLMN, 126
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_QmiNas3GppNetworkInfo, 188
nas_QmiNas3GppNetworkRAT, 189
nas_QmisNasPcsDigit, 190
nas_WCDMASysInfo, 224
unpack_nas_GetServingNetwork_t, 515
MDMCallDuration
connectionStatus, 52
MDMConnStatus
connectionStatus, 52
MEID_MAX_SIZE
dms.h, 747
MEIDString
unpack_dms_GetDeviceSerialNumbers_t, 449
MIN
unpack_dms_GetVoiceNumber_t, 461
MINREQBKLEN
common.h, 736
MMTlv
unpack_sms_SetNewSMSCallback_ind_t, 594
MNC
nas_CDMASysInfo, 121
nas_currentPLMN, 126
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_QmiNas3GppNetworkInfo, 189
nas_QmiNas3GppNetworkRAT, 189
nas_QmisNasPcsDigit, 190
nas_WCDMASysInfo, 224
unpack_nas_GetServingNetwork_t, 515
MPTlv
NASQmiCbkNasSystemSelPrefInd, 235
MSGID_AND_LEN
common.h, 736
MSGID_DONT_CARE
common.h, 736
MTMessageInfo
newMTMessageTlv, 238
manString
pack_dms_SLQSSwiSetHostDevInfo_t, 250
unpack_dms_SLQSSwiGetHostDevInfo_t, 476
manufacturer
unpack_dms_GetManufacturer_t, 454
Mask
pack_wds_SLQSGetDUNCallInfo_t, 372
mask
loc_BdsSV, 97
loc_CellDb, 98
loc_ClkInfo, 99
loc_GnssData, 100
loc_SV, 108
unpack_qos_IPv6TrafCls_t, 571
unpack_qos_Tos_t, 589
max
_PackSwiAVMSSettingsPeriodsInfo, 32
max_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 661
max_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 661
max_dist
pack_swiloc_SwiLocSetAutoStart_t, 325
unpack_swiloc_SwiLocGetAutoStart_t, 604
max_dist_reported
unpack_swiloc_SwiLocGetAutoStart_t, 604
max_time
pack_swiloc_SwiLocSetAutoStart_t, 325

unpack_swiloc_SwiLocGetAutoStart_t, 605
max_time_reported
unpack_swiloc_SwiLocGetAutoStart_t, 605
MaxAllowedPktSz
unpack_qos_swiQosFlow_t, 587
MaxChanRxRate
dunchannelRate, 60
MaxChanTxRate
dunchannelRate, 60
maxChannelRXRate
unpack_wds_GetConnectionRate_t, 646
maxChannelTXRate
unpack_wds_GetConnectionRate_t, 646
maxDlBitRate
LibPackQosClassID, 91
maxDownlinkBitrate
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
maxImages
FMSImageIDEntries, 64
MaxRXChannelRate
unpack_dms_GetDeviceCap_t, 445
maxRxChannelRate
unpack_dms_GetDeviceCapabilities_t, 446
maxSDUSize
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
maxStorageSize
sms_maxStorageSizeResp, 398
MaxTXChannelRate
unpack_dms_GetDeviceCap_t, 445
maxTxChannelRate
unpack_dms_GetDeviceCapabilities_t, 446
maxUlBitRate
LibPackQosClassID, 91
maxUplinkBitrate
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
mcc
nas_CSGID, 125
nas_MNRInfo, 169
nas_netSelectionPref, 170
nas_OperatorPLMNData, 176
pack_nas_SLQSGetPLMNName_t, 278
unpack_nas_GetHomeNetwork_t, 511
mccM
nas_minBasedIMSI, 168
mccT
nas_trueIMSI, 213
mdmCallDurationActive
unpack_wds_SLQSGetDUNCallInfo_t, 664
meanThroughputClass
LibPackGPRSRequestedQoS, 68
wds_GPRSQoS, 721
meid
unpack_dms_GetSerialNumbers_t, 459
meidSize
unpack_dms_GetDeviceSerialNumbers_t, 449
meidString
unpack_dms_SLQSSwiGetSerialNoExt_t, 478
message
unpack_sms_SLQSGetSMS_t, 596
message_type
NASOTAMessageTlv, 229
messageClass
sms_routeEntry, 401
messageFailureCode
unpack_sms_SendSMS_t, 593
messageFormat
pack_sms_SaveSMS_t, 312
pack_sms_SendSMS_t, 313
sms_sendAsyncsmsParams, 403
unpack_sms_SLQSGetSMS_t, 596
messageID
unpack_sms_SendSMS_t, 593
messageIndex
pack_sms_SLQSGetSMS_t, 315
pack_sms_SLQSModifySMSStatus_t, 317
qmiSmsMessageList, 385
sMSMTMessage, 409
messageList
unpack_sms_SLQSGetSMSList_t, 597
messageListSize
unpack_sms_SLQSGetSMSList_t, 597
messageMode
sMSMessageMode, 408
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 600
MessageModeInfo
messageModeTlv, 111
messageModeTlv, 110
MessageModeInfo, 111
TlvPresent, 111
messageSize
pack_sms_SaveSMS_t, 312
pack_sms_SendSMS_t, 313
sms_sendAsyncsmsParams, 403
unpack_sms_SLQSGetSMS_t, 596
messageTag
pack_sms_SLQSModifySMSStatus_t, 317
qmiSmsMessageList, 385
unpack_sms_SLQSGetSMS_t, 596
messageType
sms_routeEntry, 401
min
_PackSwiAVMSSettingsPeriodsInfo, 32
MinPolicedPktSz
unpack_qos_swiQosFlow_t, 587
minSize
unpack_dms_GetVoiceNumber_t, 461
minute
nas_timeInfo, 212
nas_UniversalTime, 218
mipMode
unpack_wds_GetMobileIP_t, 651
mipStatus

unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
mipstatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
mnc
nas_CSGID, 125
nas_MNRInfo, 169
nas_netSelectionPref, 170
nas_OperatorPLMNData, 176
pack_nas_SLQSGetPLMNName_t, 278
unpack_nas_GetHomeNetwork_t, 511
mncPcsDigits
nas_CSGID, 125
mobileCountryCode
sMSEtwsPlmn, 408
mobileIP
pack_wds_SLQSSetWdsEventCallback_t, 378
mobileNetworkCode
sMSEtwsPlmn, 408
mode
pack_dms_SetEventReport_t, 246
pack_dms_SetPower_t, 247
pack_loc_SetOperationMode_t, 261
pack_sms_SLQSGetSmsBroadcastConfig_t, 315
pack_sms_SLQSSetSmsBroadcastActivation_t, 318
pack_sms_SLQSSetSmsBroadcastConfig_t, 319
pack_wds_SetMobileIP_t, 366
uim_refreshevent, 430
voice_callInfo, 694
ModePref
NASModePreferenceTlv, 228
modelString
pack_dms_SLQSSwiSetHostDevInfo_t, 250
unpack_dms_SLQSSwiGetHostDevInfo_t, 476
modelid
unpack_dms_GetModelID_t, 455
modelid_str
unpack_dms_GetFirmwareInfo_t, 450
modemMode
nas_CommInfo, 124
modemindex
pack_fms_SetImagesPreference_t, 257
month
nas_timeInfo, 212
nas_UniversalTime, 218
msgCount
sms_messageWaitingInfoContent, 398
msgProtocol
sms_msgProtocolResp, 399
msgType
sms_messageWaitingInfoContent, 398
msgWaitInfo
sms_getMsgWaitingInfo, 395
msgid
pack_qmi_t, 309
unpack_qmi_t, 570
msgtype
common.h, 738
Mtu
unpack_wds_SLQSGetRuntimeSettings_t, 667
MultDisc
nas_protocolSubtypeElement, 186
multiplier
unpack_qos_pktErrRate_t, 572

NAI
unpack_wds_GetMobileIPProfile_t, 653
NAS_NAM_NAME_LENGTH
nas.h, 827
NAS_PLMN_LENGTH
nas.h, 827
NASBandPreferenceTlv, 225
band_pref, 225
TlvPresent, 225
NASEmergencyModeTlv, 225
EmerMode, 225
TlvPresent, 225
NASGWAcqOrderPrefTlv, 226
GWAcqOrderPref, 226
TlvPresent, 226
NASLTEBandPreferenceTlv, 226
LTEBandPref, 227
TlvPresent, 227
NASLteNasReleaseInfoTlv, 227
nas_major, 227
nas_minor, 227
nas_release, 227
TlvPresent, 227
NASModePreferenceTlv, 227
ModePref, 228
TlvPresent, 228
NASNetSelPreferenceTlv, 228
NetSelPref, 228
TlvPresent, 228
NASOTAMessageTlv, 228
data_buf, 229
data_len, 229
message_type, 229
TlvPresent, 229
NASPRLPreferenceTlv, 233
PRLPref, 233
TlvPresent, 233
NASPhyCaAggPcellInfo, 229
dl_bw_value, 230
freq, 230
iLTEbandValue, 230
pci, 230
TlvPresent, 230
NASPhyCaAggScellDlBw, 230
dl_bw_value, 230
TlvPresent, 230
NASPhyCaAggScellIndType, 231
freq, 232
pci, 232
scell_state, 232

TlvPresent, 232
NASPhyCaAggScellIndex, 230
scell_idx, 231
TlvPresent, 231
NASPhyCaAggScellInfo, 232
dl_bw_value, 233
freq, 233
iLTEbandValue, 233
pci, 233
scell_state, 233
TlvPresent, 233
NASQmiCbkNasSwiOTAMessageInd, 233
nasRelInfoTlv, 234
otaMsgTlv, 234
timeTlv, 234
NASQmiCbkNasSystemSelPrefInd, 234
BPTlv, 235
EMTlv, 235
GWAOPTlv, 235
LBPTlv, 235
MPTlv, 235
NSPTlv, 235
PRLPTlv, 235
RPTlv, 235
SDPTlv, 235
NASRoamPreferenceTlv, 235
RoamPref, 235
TlvPresent, 236
NASServDomainPrefTlv, 236
SrvDomainPref, 236
TlvPresent, 236
NASServingSystemInfo, 236
csAttachState, 237
hdrPersonality, 237
psAttachState, 237
radioInterfaceList, 237
radioInterfaceNo, 237
registrationState, 237
selectedNetwork, 237
NASTimeInfoTlv, 237
time, 238
TlvPresent, 238
NMEAData
unpack_loc_EventNMEA_Ind_t, 494
NSPTlv
NASQmiCbkNasSystemSelPrefInd, 235
NWQoSStatus
unpack_qos_SLQSQosGetNetworkStatus_t, 575
naiSize
unpack_wds_GetMobileIPProfile_t, 653
namID
pack_nas_SLQSNasGet3GPP2Subscription_t, 284
voice_airTimer, 676
voice_prefVoiceSO, 709
voice_roamTimer, 711
namName
nas_namName, 169
namNameLen
nas_namName, 169
Name
unpack_nas_GetServingNetwork_t, 515
name
unpack_nas_GetHomeNetwork_t, 511
unpack_wds_GetDefaultProfile_t, 649
nameLen
voice_remotePartyName, 710
namePI
voice_remotePartyName, 710
nameSize
unpack_nas_GetHomeNetwork3GPP2_t, 510
unpack_nas_GetServingNetwork_t, 515
nameString
pack_dms_SLQSSwiSetOSInfo_t, 250
unpack_dms_SLQSSwiGetOSInfo_t, 477
namelength
unpack_omaDmFotaTlv_t, 557
namesize
unpack_wds_GetDefaultProfile_t, 649
nas.h
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6, 828
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75, 828
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED, 828
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED, 828
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6, 828
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75, 828
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE, 828
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE, 828
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE, 828
nas.h, 819
NAS_PLMN_LENGTH, 827
pack_nas_GetACCOLC, 828
pack_nas_GetANAAAAuthenticationStatus, 829
pack_nas_GetCDMANetworkParameters, 829
pack_nas_GetHomeNetwork, 829

pack_nas_GetHomeNetwork3GPP2, 830
pack_nas_GetNetworkPreference, 830
pack_nas_GetRFInfo, 830
pack_nas_GetServingNetwork, 830
pack_nas_GetServingNetworkCapabilities, 831
pack_nas_GetSignalStrengths, 831
pack_nas_InitiateDomainAttach, 831
pack_nas_PerformNetworkScan, 832
pack_nas_SLQSConfigSigInfo, 834
pack_nas_SLQSGetErrorRate, 834
pack_nas_SLQSGetNetworkTime, 834
pack_nas_SLQSGetOperatorNameData, 835
pack_nas_SLQSGetPLMNName, 835
pack_nas_SLQSGetServingSystem, 835
pack_nas_SLQSGetSignalStrength, 836
pack_nas_SLQSGetSysInfo, 836
pack_nas_SLQSGetSysSelectionPref, 836
pack_nas_SLQSInitiateNetworkRegistration, 837
pack_nas_SLQSNASSwiGetChannelLock, 840
pack_nas_SLQSNASSwiSetChannelLock, 841
pack_nas_SLQSNasConfigSigInfo2, 837
pack_nas_SLQSNasGet3GPP2Subscription, 837
pack_nas_SLQSNasGetCellLocationInfo, 838
pack_nas_SLQSNasGetHDRColorCode, 838
pack_nas_SLQSNasGetSigInfo, 838
pack_nas_SLQSNasGetTxRxInfo, 839
pack_nas_SLQSNasIndicationRegisterExt, 839
pack_nas_SLQSNasSwiIndicationRegister, 840
pack_nas_SLQSNasSwiModemStatus, 840
pack_nas_SLQSSetBandPreference, 841
pack_nas_SLQSSetSignalStrengthsCallback, 841
pack_nas_SLQSSetSysSelectionPref, 842
pack_nas_SLQSSwiGetHDRPersonality, 842
pack_nas_SLQSSwiGetHDRProtSubtype, 842
pack_nas_SLQSSwiGetHRPDStats, 843
pack_nas_SLQSSwiGetLteCQI, 843
pack_nas_SLQSSwiGetLteSccRxInfo, 843
pack_nas_SLQSSwiNetworkDebug, 844
pack_nas_SLQSSwiPSDetach, 844
pack_nas_SetACCOLC, 832
pack_nas_SetCDMANetworkParameters, 832
pack_nas_SetLURejectCallback, 833
pack_nas_SetNetworkPreference, 833
pack_nas_SetRFInfoCallback, 833
pack_nas_SlqsGetLTECphyCAInfo, 834
unpack_nas_GetACCOLC, 844
unpack_nas_GetANAAAAuthenticationStatus, 845
unpack_nas_GetCDMANetworkParameters, 845
unpack_nas_GetHomeNetwork, 845
unpack_nas_GetHomeNetwork3GPP2, 846
unpack_nas_GetNetworkPreference, 846
unpack_nas_GetRFInfo, 846
unpack_nas_GetServingNetwork, 847
unpack_nas_GetServingNetworkCapabilities, 847
unpack_nas_GetSignalStrengths, 847
unpack_nas_InitiateDomainAttach, 848
unpack_nas_PerformNetworkScan, 848
unpack_nas_SLQSConfigSigInfo, 851
unpack_nas_SLQSGetErrorRate, 852
unpack_nas_SLQSGetNetworkTime, 852
unpack_nas_SLQSGetOperatorNameData, 853
unpack_nas_SLQSGetPLMNName, 853
unpack_nas_SLQSGetServingSystem, 853
unpack_nas_SLQSGetSignalStrength, 854
unpack_nas_SLQSGetSysInfo, 854
unpack_nas_SLQSGetSysSelectionPref, 854
unpack_nas_SLQSInitiateNetworkRegistration, 855
unpack_nas_SLQSNASSwiGetChannelLock, 858
unpack_nas_SLQSNASSwiSetChannelLock, 859
unpack_nas_SLQSNasConfigSigInfo2, 855
unpack_nas_SLQSNasGet3GPP2Subscription, 855
unpack_nas_SLQSNasGetCellLocationInfo, 856
unpack_nas_SLQSNasGetHDRColorCode, 856
unpack_nas_SLQSNasGetSigInfo, 856
unpack_nas_SLQSNasGetTxRxInfo, 857
unpack_nas_SLQSNasIndicationRegisterExt, 857
unpack_nas_SLQSNasNetworkTimeCallBack_ind, 857
unpack_nas_SLQSNasSigInfoCallback_ind, 858
unpack_nas_SLQSNasSwiIndicationRegister, 858
unpack_nas_SLQSNasSwiModemStatus, 859
unpack_nas_SLQSNasSwiOTAMessageCallback-_ind, 859
unpack_nas_SLQSNasSysInfoCallback_ind, 860
unpack_nas_SLQSNasTimerCallback_ind, 860
unpack_nas_SLQSSetBandPreference, 860
unpack_nas_SLQSSetSignalStrengthsCallback, 860
unpack_nas_SLQSSetSysSelectionPref, 861
unpack_nas_SLQSSetSysSelectionPrefCallBack-_ind, 861
unpack_nas_SLQSSwiGetHDRPersonality, 861
unpack_nas_SLQSSwiGetHDRProtSubtype, 862
unpack_nas_SLQSSwiGetHRPDStats, 862
unpack_nas_SLQSSwiGetLteCQI, 862
unpack_nas_SLQSSwiGetLteSccRxInfo, 863
unpack_nas_SLQSSwiNetworkDebug, 863
unpack_nas_SLQSSwiPSDetach, 863
unpack_nas_SetACCOLC, 848
unpack_nas_SetCDMANetworkParameters, 849
unpack_nas_SetDataCapabilitiesCallback_ind, 849
unpack_nas_SetEventReportInd, 849
unpack_nas_SetLURejectCallback, 849
unpack_nas_SetNasLTECphyCaIndCallback_ind, 850
unpack_nas_SetNetworkPreference, 850
unpack_nas_SetRFInfoCallback, 850
unpack_nas_SetRoamingIndicatorCallback_ind, 851
unpack_nas_SetServingSystemCallback_ind, 851
unpack_nas_SlqsGetLTECphyCAInfo, 852
nas_ActPilotPNElement, 111
ActSetPilotPN, 112

ActSetPilotPNStrength, 112
nas_AddCDMASysInfo, 112
geoSysIdx, 112
regPrd, 112
nas_AddSysInfo, 112
cellBroadcastCap, 113
geoSysIdx, 113
nas_CDMAChannel, 115
priChA, 115
priChB, 115
secChA, 115
secChB, 115
nas_CDMAECIOThresh, 115
CDMAECIOThreshListLen, 116
pCDMAECIOThreshList, 116
nas_CDMAInfo, 116
baseId, 117
baseLat, 117
baseLong, 117
nid, 117
refpn, 117
sid, 117
nas_CDMARSSIThresh, 117
CDMARSSIThreshListLen, 117
pCDMARSSIThreshList, 117
nas_CDMASysInfo, 118
baseId, 120
baseLat, 120
baseLong, 121
bsInfoValid, 121
bsPRev, 121
bsPRevValid, 121
ccsSupported, 121
ccsSupportedValid, 121
cdmaSysIdValid, 121
isSysPrlMatch, 121
isSysPrlMatchValid, 121
MCC, 121
MNC, 121
networkID, 121
networkIdValid, 121
pRevInUse, 121
pRevInUseValid, 121
packetZone, 121
packetZoneValid, 121
sysInfoCDMA, 121
systemID, 121
nas_CDMASysInfoExt, 121
imsi_11_12, 122
MCC, 122
nas_CSGID, 124
id, 125
mcc, 125
mnc, 125
mncPcsDigits, 125
rat, 125
nas_CallBarringSysInfo, 113
csBarStatus, 114
psBarStatus, 114
nas_CommInfo, 123
imsRegState, 124
modemMode, 124
psState, 124
systemMode, 124
temperature, 124
nas_DRCParams, 132
DRCCover, 132
DRCValue, 132
nas_DataStatusDetail, 127
IPAddress, 129
LastErrCode, 129
nas_DeviceConfigDetail, 130
Chipset, 131
HWVersion, 131
QLIC, 131
Technology, 131
nas_ECIOThresh, 133
ECIOThresListLen, 133
pECIOThresList, 133
nas_GERANInfo, 134
arfcn, 136
bsic, 136
cellID, 136
insNmrCellInfo, 136
lac, 136
nmrInst, 136
plmn, 136
rxLev, 136
timingAdvance, 136
nas_GSMRSSIThresh, 138
GSMRSSIThreshListLen, 138
pGSMRSSIThreshList, 138
nas_GSMSrvStatusInfo, 138
isPrefDataPath, 139
srvStatus, 139
trueSrvStatus, 139
nas_GSMSysInfo, 139
cellId, 141
cellIdValid, 141
dtmSupp, 141
dtmSuppValid, 141
egprsSupp, 142
egprsSuppValid, 142
lac, 142
lacValid, 142
MCC, 142
MNC, 142
networkIdValid, 142
regRejectInfoValid, 142
rejCause, 142
rejectSrvDomain, 142
sysInfoGSM, 142
nas_HDRECIOThresh, 142
HDRECIOThreshListLen, 142
pHDRECIOThreshList, 142
nas_HDRIOThresh, 142

HDRIOThreshListLen, 143
pHDRIOThreshList, 143
nas_HDRRSSIThresh, 143
HDRRSSIThreshListLen, 143
pHDRRSSIThreshList, 143
nas_HDRSINRThresh, 144
HDRSINRThresListLen, 144
pHDRSINRThresList, 144
nas_HDRSINRThreshold, 144
HDRSINRThreshListLen, 145
pHDRSINRThreshList, 145
nas_HDRSysInfo, 145
hdrActiveProt, 147
hdrActiveProtValid, 147
hdrPersonality, 147
hdrPersonalityValid, 147
is856SysId, 147
is856SysIdValid, 147
isSysPrlMatch, 147
isSysPrlMatchValid, 147
sysInfoHDR, 147
nas_IOThresh, 149
IOThresListLen, 149
pIOThresList, 149
nas_LTEInfo, 151
band, 153
bandwidth, 153
emmConnState, 153
emmState, 153
emmSubState, 153
RXChan, 153
TXChan, 153
nas_LTEInfoInterfreq, 154
freqsLen, 154
InfoInterfreq, 154
ueInIdle, 154
nas_LTEInfoIntrafreq, 154
CellParams, 156
cellReselPriority, 156
cellsLen, 156
earfcn, 156
globalCellId, 156
plmn, 156
sIntraSearch, 156
sNonIntraSearch, 156
servingCellId, 156
tac, 156
threshServingLow, 156
ueInIdle, 156
nas_LTEInfoNeighboringGSM, 156
freqsLen, 157
LteGsmCellInfo, 157
ueInIdle, 157
nas_LTEInfoNeighboringWCDMA, 157
freqsLen, 158
ueInIdle, 158
nas_LTERSRPThresh, 159
LTERSRPThreshListLen, 159
pLTERSRPThreshList, 159
nas_LTERSRQThresh, 159
LTERSRQThreshListLen, 160
pLTERSRQThreshList, 160
nas_LTERSSIThresh, 160
LTERSSIThreshListLen, 160
pLTERSSIThreshList, 160
nas_LTESNRThresh, 163
LTESNRThresListLen, 163
pLTESNRThresList, 163
nas_LTESNRThreshold, 163
LTESNRThreshListLen, 164
pLTESNRThreshList, 164
nas_LTESigRptCfg, 160
avgPeriod, 161
rptRate, 161
nas_LTESigRptConfig, 161
avgPeriod, 162
rptRate, 162
nas_LTESysInfo, 164
cellId, 166
cellIdValid, 166
lac, 166
lacValid, 166
MCC, 166
MNC, 166
networkIdValid, 166
regRejectInfoValid, 166
rejCause, 166
rejectSrvDomain, 166
sysInfoLTE, 166
tac, 166
tacValid, 166
nas_MNRInfo, 168
mcc, 169
mnc, 169
rat, 169
nas_NetworkStat1x, 170
ActSetCnt, 172
NeighborSetCnt, 172
pActPilotPNElements, 172
pNeighborSetPilotPN, 172
RX_EC_IO, 172
RX_PWR, 172
SO, 172
State, 172
TX_PWR, 172
nas_NetworkStatEVDO, 172
MACIndex, 173
PER, 174
pSectorID, 174
PilotEnergy, 174
RX_PWR, 174
SNR, 174
SectorIDLen, 174
State, 174
nas_OperatorPLMNData, 175
lac1, 176

lac2, 176
mcc, 176
mnc, 176
PLMNRecID, 176
nas_PLMNNetworkName, 183
numInstance, 183
PLMNNetName, 183
nas_PLMNNetworkNameData, 184
codingScheme, 185
countryInitials, 185
longName, 185
longNameLen, 185
longNameSpareBits, 185
shortName, 185
shortNameLen, 185
shortNameSpareBits, 185
nas_PhyCaAggPcellInfo, 177
dl_bw_value, 177
freq, 177
iLTEbandValue, 177
pci, 178
TlvPresent, 178
nas_PhyCaAggScellDlBw, 178
dl_bw_value, 178
TlvPresent, 178
nas_PhyCaAggScellIndType, 179
freq, 179
pci, 179
scell_state, 179
TlvPresent, 179
nas_PhyCaAggScellIndex, 178
scell_idx, 179
TlvPresent, 179
nas_PhyCaAggScellInfo, 179
dl_bw_value, 182
freq, 182
iLTEbandValue, 182
pci, 182
scell_state, 182
TlvPresent, 182
nas_PilotSetData, 182
NumPilots, 182
pPilotSetInfo, 182
nas_PilotSetParams, 182
PilotPN, 183
PilotStrength, 183
PilotType, 183
nas_QmiNas3GppNetworkInfo, 187
Desription, 188
Forbidden, 188
InUse, 188
MCC, 188
MNC, 189
Preferred, 189
Roaming, 189
nas_QmiNas3GppNetworkRAT, 189
MCC, 189
MNC, 189
RAT, 189
nas_QmisNasPcsDigit, 189
includes_pcs_digit, 190
MCC, 190
MNC, 190
nas_RFInfoTlv, 190
activeBandClass, 191
activeChannel, 191
radioInterface, 191
radioInterfaceSize, 191
TlvPresent, 191
nas_RSRPThresh, 192
pRSRPThresList, 193
RSRPThresListLen, 193
nas_RSRQThresh, 193
pRSRQThresList, 194
RSRQThresListLen, 194
nas_RSSIThresh, 194
pRSSIThresList, 194
RSSIThresListLen, 194
nas_RejectReasonTlv, 190
rejectCause, 190
serviceDomain, 190
TlvPresent, 190
nas_RxSigInfo, 196
isRadioTuned, 196
rsrp, 196
rxChainIndex, 196
rxPower, 197
nas_SLQSSignalStrengthsIndReq, 201
ecioDelta, 203
ecioThresholdList, 203
ecioThresholdListLen, 203
ioDelta, 203
lteRsrpDelta, 203
lteSnrDelta, 203
rsrqDelta, 203
rxSignalStrengthDelta, 203
sinrDelta, 203
sinrThresholdList, 203
sinrThresholdListLen, 203
nas_SLQSSignalStrengthsInformation, 203
ecioInfo, 204
errorRateInfo, 204
io, 204
lteRsrpinfo, 204
lteSnrinfo, 204
rsrqInfo, 204
rxSignalStrengthInfo, 204
sinr, 204
nas_SLQSSignalStrengthsTlv, 204
sSLQSSignalStrengthsInfo, 204
TlvPresent, 204
nas_SccRxInfo, 197
numInstances, 198
rsrq, 198
sigInfo, 198
snr, 198

TlvPresent, 198
nas_SignalStrengthTlv, 201
radioInterface, 201
signalStrength, 201
TlvPresent, 201
nas_SrvStatusInfo, 205
isPrefDataPath, 205
srvStatus, 205
nas_TDSCDMAECIOThresh, 208
nas_TDSCDMARSCPThresh, 208
nas_TDSCDMARSSIThresh, 209
nas_TDSCDMASINRCONFThresh, 209
nas_TDSCDMASINRThresh, 210
nas_UMTSInfo, 213
cellID, 214
ecio, 215
geranInst, 215
GeranInstInfo, 215
lac, 215
plmn, 215
psc, 215
rscp, 215
UMTSInstInfo, 215
uarfcn, 215
umtsInst, 215
nas_UMTSinstInfo, 215
umtsEcio, 215
umtsPsc, 215
umtsRscp, 216
umtsUarfcn, 216
nas_UniversalTime, 217
day, 218
dayOfWeek, 218
hour, 218
minute, 218
month, 218
second, 218
year, 218
nas_WCDMAECIOThresh, 219
nas_WCDMAInfoLTENeighborCell, 219
umtsLTENbrCellLen, 220
wcdmaRRCState, 220
nas_WCDMARSSIThresh, 220
nas_WCDMASysInfo, 221
cellId, 223
cellIdValid, 223
hsCallStatus, 223
hsCallStatusValid, 223
hsInd, 224
hsIndValid, 224
lac, 224
lacValid, 224
MCC, 224
MNC, 224
networkIdValid, 224
psc, 224
pscValid, 224
regRejectInfoValid, 224
rejCause, 224
rejectSrvDomain, 224
sysInfoWCDMA, 224
nas_acqOrderPref, 111
acqOrdeLen, 111
pAcqOrder, 111
nas_callBarStatus, 114
csBarStatus, 115
psBarStatus, 115
nas_cellParams, 122
pci, 123
rsrp, 123
rsrq, 123
rssi, 123
srxlev, 123
nas_currentPLMN, 125
MCC, 126
MNC, 126
netDescr, 126
netDescrLength, 126
nas_dataSrvCapabilities, 126
dataCapabilities, 127
dataCapabilitiesLen, 127
nas_detailSvcInfo, 129
hdrHybrid, 130
hdrSrvStatus, 130
isSysForbidden, 130
srvCapability, 130
srvStatus, 130
nas_dirNum, 131
dirNum, 131
dirNumLen, 132
nas_ecioListElement, 132
ecio, 133
radioIf, 133
nas_errorRateListElement, 133
errorRate, 134
radioIf, 134
nas_geranInstInfo, 136
geranArfcn, 136
geranBsicBcc, 136
geranBsicNcc, 137
geranRssi, 137
nas_gsmCellInfo, 137
arfcn, 137
band1900, 138
bsicId, 138
cellIdValid, 138
rssi, 138
srxlev, 138
nas_homeSIDNID, 147
numInstances, 147
SidNid, 147
nas_infoInterFreq, 148
cell_resel_priority, 148
cellInterFreqParams, 148
cells_len, 148
earfcn, 148

threshXHigh, 149
threshXLow, 149
nas_lteEARFCN, 149
earfcn0, 150
earfcn1, 150
status, 150
nas_lteGsmCellInfo, 150
cellReselPriority, 151
cells_len, 151
GsmCellInfo, 151
nccPermitted, 151
threshGsmHigh, 151
threshGsmLow, 151
nas_ltePCI, 158
earfcn, 158
pci, 158
status, 158
nas_lteRsrpinformation, 158
rsrplevel, 159
nas_lteSnrinformation, 162
snrlevel, 162
nas_lteWcdmaCellInfo, 166
cellReselPriority, 167
cellsLen, 167
threshXhigh, 167
threshXlow, 167
uarfcn, 167
WCDMACellInfo, 167
nas_major
NASLteNasReleaseInfoTlv, 227
nas_minBasedIMSI, 167
imsiM1112, 168
imsiMS1, 168
imsiMS2, 168
mccM, 168
nas_minor
NASLteNasReleaseInfoTlv, 227
nas_namName, 169
namName, 169
namNameLen, 169
nas_netSelectionPref, 169
mcc, 170
mnc, 170
netReg, 170
nas_nmrCellInfo, 174
nmrArfcn, 175
nmrBsic, 175
nmrCellID, 175
nmrLac, 175
nmrPlmn, 175
nmrRxLev, 175
nas_operatorNameString, 175
PLMNName, 175
nas_operatorPLMNList, 176
numInstance, 177
PLMNData, 177
nas_protocolSubtypeElement, 185
AccessMac, 186
AuthProt, 186
ControlMac, 186
EncryptProt, 186
ForwardMac, 186
IdleState, 186
KeyExchange, 186
MultDisc, 186
PhysicalLayer, 187
ReverseMac, 187
SecProt, 187
VirtStream, 187
nas_qaQmi3Gpp2TimeZone, 187
daylightSavings, 187
leapSeconds, 187
localTimeOffset, 187
nas_release
NASLteNasReleaseInfoTlv, 227
nas_roamIndList, 191
numInstances, 192
radioInterface, 192
roamIndicator, 192
nas_rsrqInformation, 193
radioIf, 193
rsrq, 193
nas_rxInfo, 195
ecio, 195
isRadioTuned, 195
phase, 195
rscp, 195
rsrp, 195
rxPower, 196
nas_rxSignalStrengthListElement, 197
radioIf, 197
rxSignalStrength, 197
nas_servSystem, 199
csAttachState, 200
numRadioInterfaces, 200
psAttachState, 200
radioInterface, 200
regState, 200
selNetwork, 200
nas_serviceProviderName, 198
displayCondition, 199
spn, 199
spnLength, 199
nas_sidNid, 201
nid, 201
sid, 201
nas_sysInfoCommon, 205
isSysForbidden, 207
isSysForbiddenValid, 207
roamStatus, 207
roamStatusValid, 207
srvCapability, 207
srvCapabilityValid, 207
srvDomain, 207
srvDomainValid, 207
nas_timeInfo, 210

day, 212
dayLtSavingAdj, 212
dayOfWeek, 212
hour, 212
minute, 212
month, 212
radioInterface, 212
second, 212
timeZone, 212
TlvPresent, 212
year, 212
nas_trueIMSI, 212
imsiT1112, 213
imsiTS1, 213
imsiTS2, 213
imsiTaddrNum, 213
mccT, 213
nas_txInfo, 213
isInTraffic, 213
txPower, 213
nas_umtsLTENbrCell, 216
cellIsTDD, 216
earfcn, 216
pci, 216
rsrp, 216
rsrq, 217
srxlev, 217
nas_wcdmaCellInfo, 218
cpich_ecno, 218
cpich_rscp, 218
psc, 219
srxlev, 219
nas_wcdmaUARFCN, 224
status, 224
uarfcn, 224
NasGetLTECphyCaInfo, 225
PhyCaAggPcellInfo, 226
PhyCaAggScellDlBw, 226
PhyCaAggScellIndType, 226
PhyCaAggScellIndex, 226
PhyCaAggScellInfo, 226
nasRelInfoTlv
NASQmiCbkNasSwiOTAMessageInd, 234
nccPermitted
nas_lteGsmCellInfo, 151
NeighborSetCnt
nas_NetworkStat1x, 172
netDescr
nas_currentPLMN, 126
netDescrLength
nas_currentPLMN, 126
netInfoLen
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
netReg
nas_netSelectionPref, 170
NetSelPref
NASNetSelPreferenceTlv, 228
NetworkID
unpack_nas_SLQSGetServingSystem_t, 528
networkID
nas_CDMASysInfo, 121
networkIdValid
nas_CDMASysInfo, 121
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_WCDMASysInfo, 224
networkInfoLen
unpack_wds_SLQSGetCurrDataSystemStat_t, 660
NetworkType
currNetworkInfo, 55
wds_currNetworkInfo, 714
NewMMTlv
unpack_sms_SetNewSMSCallback_ind_t, 594
newMTMessageTlv, 238
MTMessageInfo, 238
TlvPresent, 238
newPINLen
uim_unblockUIMPIN, 436
newPINVal
uim_unblockUIMPIN, 436
newPasswd
pack_voice_SLQSVoiceSetCallBarringPassword_t, 354
newPasswdAgain
pack_voice_SLQSVoiceSetCallBarringPassword_t, 354
newPin
pack_dms_UIMUnblockPIN_t, 254
newValue
pack_dms_UIMChangePIN_t, 251
nextHeader
LibPackTFTIDParams, 93
nid
nas_CDMAInfo, 117
nas_sidNid, 201
unpack_nas_GetHomeNetwork_t, 511
nmrArfcn
nas_nmrCellInfo, 175
nmrBsic
nas_nmrCellInfo, 175
nmrCellID
nas_nmrCellInfo, 175
nmrInst
nas_GERANInfo, 136
nmrLac
nas_nmrCellInfo, 175
nmrPlmn
nas_nmrCellInfo, 175
nmrRxLev
nas_nmrCellInfo, 175
noReplyTimer
voice_callFWExtInfo, 691
voice_callFWInfo, 692
notification

unpack_omaDmNotificationsTlv_t, 558
notificationType
sMSEtwsMessage, 407
notused
unpack_dms_SetCrashAction_t, 462
num_instances
DMScustSettingList, 59
sms_qaQmi3GPP2BroadcastCfgInfo, 400
sms_qaQmi3GPPBroadcastCfgInfo, 400
numApp
slotInf, 392
uim_slotInfo, 435
numCrashes
crashInformation, 54
numEntries
unpack_dms_SLQSSwiGetFirmwareCurr_t, 474
numFeatures
uim_personalizationStatus, 427
numFiles
uim_registerRefresh, 431
NumFlows
unpack_qos_SLQSSetQosEventCallback_ind_t, 578
numInstance
nas_operatorPLMNList, 177
nas_PLMNNetworkName, 183
numInstances
nas_homeSIDNID, 147
nas_roamIndList, 192
nas_SccRxInfo, 198
sms_getMsgWaitingInfo, 395
voice_arrAlertingPattern, 679
voice_arrAlertingType, 680
voice_arrAlphaID, 681
voice_arrCalledPartyNum, 681
voice_arrCallEndReason, 682
voice_arrCallInfo, 682
voice_arrConnectPartyNum, 683
voice_arrDiagInfo, 684
voice_arrRedirPartyNum, 684
voice_arrRemotePartyName, 685
voice_arrRemotePartyNum, 685
voice_arrSvcOption, 686
voice_arrUUSInfo, 686
voice_getCallFWExtInfo, 704
voice_getCallFWInfo, 705
wds_DomainNameList, 720
wds_PCSCFFQDNAddressList, 724
wds_PCSCFIPv4ServerAddressList, 724
numLen
voice_callFWExtInfo, 691
voice_callFWInfo, 692
voice_peerNumberInfo, 707
voice_remotePartyNum, 710
numOfFiles
uim_refreshevent, 430
numOfRoutes
sms_setRoutesReq, 405
numOpt
wds_DHCPLeaseOptTlv, 715
wds_DHCPv4OptionList, 719
wdsDhcpv4OptionList, 730
numPI
voice_peerNumberInfo, 707
NumPilots
nas_PilotSetData, 182
numPlan
voice_callFWExtInfo, 691
voice_connectNumInfo, 699
voice_peerNumberInfo, 707
numPresInd
voice_connectNumInfo, 699
numQosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 577
numRadioInterfaces
nas_servSystem, 200
NumRxFilters
unpack_qos_QosFlowInfo_t, 574
numSI
voice_peerNumberInfo, 707
numSlot
uim_cardStatus, 419
NumSupUSBComps
unpack_dms_GetUSBComp_t, 460
NumTxFilters
unpack_qos_QosFlowInfo_t, 574
numType
voice_callFWExtInfo, 691
voice_connectNumInfo, 700
voice_peerNumberInfo, 707
number
voice_callFWExtInfo, 691
voice_callFWInfo, 692
voice_peerNumberInfo, 707
numberPlan
voice_callFwdTypeAndPlan, 689
numberType
voice_callFwdTypeAndPlan, 689
NxtHdrProto
unpack_qos_swiQosFilter_t, 583

OKtoRefresh
pack_uim_SLQSUIMRefreshOK_t, 337
OMADMEnabled
unpack_swiavms_SLQSAVMSGetSettings_t, 601
unpack_swioma_SLQSOMADMGetSettings_t, 609
oddEvenInd
voice_calledPartySubAdd, 688
OfflineReason
unpack_dms_GetPower_t, 457
offset
uim_readTransparentInfo, 428
oldPINLen
uim_changeUIMPIN, 420
oldPINVal
uim_changeUIMPIN, 420
oldPasswd

pack_voice_SLQSVoiceSetCallBarringPassword_t, 354
oldValue
pack_dms_UIMChangePIN_t, 251
opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 269
operatingMode
dms_OperatingModeTlv, 57
OperatingModeTlv
unpack_dms_SetEventReport_ind_t, 464
operation
pack_pds_SetPDSDefaults_t, 300
uim_depersonalizationInformation, 421
OperationMode
unpack_dms_GetPower_t, 457
optCode
wds_DHCPOpt, 717
wds_DHCPv4Option, 718
wdsDhcpv4Option, 729
optList
wds_DHCPLeaseOptTlv, 715
optListData
wds_DHCPLeaseOptTlv, 715
optVal
wds_DHCPv4Option, 718
wdsDhcpv4Option, 729
optValLen
wds_DHCPOpt, 717
wds_DHCPv4Option, 718
wdsDhcpv4Option, 729
otaMsgTlv
NASQmiCbkNasSwiOTAMessageInd, 234

p1Status
unpack_dms_UIMGetPINStatus_t, 482
p1UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 482
p1VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 482
p2Status
unpack_dms_UIMGetPINStatus_t, 482
p2UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 482
p2VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 482
p3GPP2TimeInfo
unpack_nas_SLQSGetNetworkTime_t, 523
p3GPPTimeInfo
unpack_nas_SLQSGetNetworkTime_t, 523
p3GppNetworkInfoInstances
unpack_nas_PerformNetworkScan_t, 517
p3GppNetworkInstanceSize
unpack_nas_PerformNetworkScan_t, 518
p3gppRelease
pack_wds_SLQSSet3GPPConfigItem_t, 376
pAAASPI
pack_wds_SetMobileIPProfile_t, 369
PACK_WDS_IPV4
wds.h, 1046
PACK_WDS_IPV6
wds.h, 1046
pAMRStatus
pack_voice_SLQSVoiceGetConfig_t, 350
pAPN
_PackSwiAVMSSettingsAPNInfo, 31
pAPNClass
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pAPNClass3GPP2
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pAPNDisabledFlag
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pAPNEnabled3GPP2
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pAPNInfo
pack_swiavms_SLQSAVMSSetSettings_t, 323
unpack_swiavms_SLQSAVMSGetSettings_t, 601
pAPNName
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pAPNnameSize
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pAccuracy
unpack_pds_GetPDSDefaults_t, 559
pAcqOrder
nas_acqOrderPref, 111
pAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 296
pActPilotPNElements
nas_NetworkStat1x, 172
pAddCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 533
unpack_nas_SLQSSysInfoCallback_ind_t, 553
pAddGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 533
unpack_nas_SLQSSysInfoCallback_ind_t, 553
pAddHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 533
unpack_nas_SLQSSysInfoCallback_ind_t, 553
pAddLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 533
unpack_nas_SLQSSysInfoCallback_ind_t, 553
pAddWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 533
unpack_nas_SLQSSysInfoCallback_ind_t, 553
pAddrAllocPref
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pAddress
pack_wds_SetMobileIPProfile_t, 369
pAirTimer
pack_voice_SLQSVoiceGetConfig_t, 350
pAirTimerCnt

unpack_voice_SLQSVoiceGetConfig_t, 638
pAirTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 355
pAirTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 642
pAlertType
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pAlertingPattern
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pAllowLinger
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pAlphaIDInfo
unpack_voice_SLQSOriginateUSSD_t, 619
unpack_voice_SLQSVoiceDialCall_t, 621
unpack_voice_SLQSVoiceGetCallBarring_t, 625
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 627
unpack_voice_SLQSVoiceGetCallInfo_t, 629
unpack_voice_SLQSVoiceGetCallWaiting_t, 630
unpack_voice_SLQSVoiceGetCLIP_t, 632
unpack_voice_SLQSVoiceGetCLIR_t, 633
unpack_voice_SLQSVoiceGetCNAP_t, 634
unpack_voice_SLQSVoiceGetCOLP_t, 635
unpack_voice_SLQSVoiceGetCOLR_t, 637
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 640
unpack_voice_SLQSVoiceSetSUPSService_t, 643
pAltitudeAssumed
unpack_loc_PositionRpt_Ind_t, 501
pAltitudeWrtEllipsoid
pack_pds_SLQSPDSInjectPositionData_t, 305
unpack_loc_BestAvailPos_Ind_t, 490
unpack_loc_PositionRpt_Ind_t, 501
pAltitudeWrtMeanSeaLevel
unpack_loc_BestAvailPos_Ind_t, 490
unpack_loc_PositionRpt_Ind_t, 501
pAltitudeWrtSealevel
pack_pds_SLQSPDSInjectPositionData_t, 305
pApnString
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pApnStringSize
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pApnname
pack_wds_SetDefaultProfile_t, 365
pAppName
loc_LocApplicationInfo, 103
pAppPriority
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pAppProvider
loc_LocApplicationInfo, 103
pAppSubType
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 548
pAppType
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pAppVersion
loc_LocApplicationInfo, 103
pApplication
pack_nas_SetCDMANetworkParameters_t, 274
pApplicationInfo
pack_loc_Start_t, 271
pArrAlertingPattern
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrAlertingType
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrAlphaID
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrCallEndReason
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrCallInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrCalledPartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrConnectPartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrDiagInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrRedirPartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrRemotePartyName
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrRemotePartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrSvcOption
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pArrUUSInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
pAuth
pack_wds_SLQSStartDataSession_t, 380
pAuthPassword
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pAuthPassword_tSize
LibPackprofile_3GPP2, 89
pAuthPasswordSize
LibpackProfile3GPP2, 78
pAuthProtocol
LibpackProfile3GPP2, 78
LibPackprofile_3GPP2, 89
pAuthRetryCount
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pAuthTimeout
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pAuthenticateResult
unpack_uim_SLQSUIMAuthenticate_t, 614
pAuthenticationPref
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pAutoAnsStatus
unpack_voice_SLQSVoiceSetConfig_t, 642

pAutoAnswer
pack_voice_SLQSVoiceGetConfig_t, 350
pack_voice_SLQSVoiceSetConfig_t, 355
pAutoAnswerStat
unpack_voice_SLQSVoiceGetConfig_t, 639
pAutoSelection
unpack_uim_SLQSUIMGetConfiguration_t, 616
pAutosdm
pack_swioma_SLQSOMADMSetSettings_t, 328
pBandPref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pBdsSVInfo
pack_loc_Delete_Assist_Data_t, 257
pBinaryUpdateSessionInfo
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
_unpack_swiavms_SLQSAVMSSessionGetInfo_t, 34
pBroadcast
pack_nas_SetCDMANetworkParameters_t, 274
pBroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 319
unpack_sms_SLQSGetSmsBroadcastConfig_t, 597
pBurstDTMFLengths
pack_voice_SLQSVoiceBurstDTMF_t, 344
pCCResType
unpack_voice_SLQSVoiceGetCallBarring_t, 625
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 627
unpack_voice_SLQSVoiceGetCallWaiting_t, 631
unpack_voice_SLQSVoiceGetCLIP_t, 632
unpack_voice_SLQSVoiceGetCLIR_t, 633
unpack_voice_SLQSVoiceGetCNAP_t, 634
unpack_voice_SLQSVoiceGetCOLP_t, 636
unpack_voice_SLQSVoiceGetCOLR_t, 637
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 641
pCCResultType
unpack_voice_SLQSVoiceDialCall_t, 621
unpack_voice_SLQSVoiceSetSUPSService_t, 643
pCCSUPSType
unpack_voice_SLQSVoiceDialCall_t, 621
unpack_voice_SLQSVoiceGetCallBarring_t, 625
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 627
unpack_voice_SLQSVoiceGetCallWaiting_t, 631
unpack_voice_SLQSVoiceGetCLIP_t, 632
unpack_voice_SLQSVoiceGetCLIR_t, 633
unpack_voice_SLQSVoiceGetCNAP_t, 634
unpack_voice_SLQSVoiceGetCOLP_t, 636
unpack_voice_SLQSVoiceGetCOLR_t, 637
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 641
unpack_voice_SLQSVoiceSetSUPSService_t, 643
pCCSuppsType
unpack_voice_SLQSOriginateUSSD_t, 620
pCDMABroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 319
unpack_sms_SLQSGetSmsBroadcastConfig_t, 597
pCDMAChannel
unpack_nas_SLQSNasGet3GPP2Subscription_t, 539
pCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 282
pCDMAECIOThreshList
nas_CDMAECIOThresh, 116
pCDMAFrameErrRate
unpack_nas_SLQSGetErrorRate_t, 522
pCDMAInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 282
pCDMARSSIThreshList
nas_CDMARSSIThresh, 117
pCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 544
pCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 553
pCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 553
pCLIPResp
unpack_voice_SLQSVoiceGetCLIP_t, 632
pCLIRResp
unpack_voice_SLQSVoiceGetCLIR_t, 633
pCLIRType
pack_voice_SLQSVoiceDialCall_t, 345
pCNAPResp
unpack_voice_SLQSVoiceGetCNAP_t, 634
pCOLPResp
unpack_voice_SLQSVoiceGetCOLP_t, 636
pCOLRResp
unpack_voice_SLQSVoiceGetCOLR_t, 637
PCSCFAddrPCO
unpack_wds_SLQSGetRuntimeSettings_t, 667
PCSCFFQDNAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 667
pCSGID
pack_nas_SLQSSetSysSelectionPref_t, 296
pCUGInfo
pack_voice_SLQSVoiceDialCall_t, 345
pCallBarringPasswd
pack_voice_SLQSVoiceSetSUPSService_t, 358
pCallForwardingNumber
pack_voice_SLQSVoiceSetSUPSService_t, 358
pCallFwdTypeAndPlan
pack_voice_SLQSVoiceSetSUPSService_t, 358
pCallID

pack_voice_SLQSVoiceManageCalls_t, 351
pack_voice_SLQSVoiceSendFlash_t, 353
pack_voice_SLQSVoiceStartContDTMF_t, 358
unpack_voice_SLQSVoiceBurstDTMF_t, 620
unpack_voice_SLQSVoiceDialCall_t, 621
unpack_voice_SLQSVoiceGetCallBarring_t, 625
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 627
unpack_voice_SLQSVoiceGetCallWaiting_t, 631
unpack_voice_SLQSVoiceGetCLIP_t, 632
unpack_voice_SLQSVoiceGetCLIR_t, 633
unpack_voice_SLQSVoiceGetCNAP_t, 634
unpack_voice_SLQSVoiceGetCOLP_t, 635
unpack_voice_SLQSVoiceGetCOLR_t, 637
unpack_voice_SLQSVoiceSendFlash_t, 640
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 641
unpack_voice_SLQSVoiceSetSUPSService_t, 643
unpack_voice_SLQSVoiceStartContDTMF_t, 644
voice_burstDTMFInfo, 687

pCallId
pack_voice_SLQSVoiceAnswerCall_t, 343
pack_voice_SLQSVoiceEndCall_t, 346
unpack_voice_SLQSOriginateUSSD_t, 619
unpack_voice_SLQSVoiceAnswerCall_t, 620
unpack_voice_SLQSVoiceEndCall_t, 622

pCallInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 629

pCallPartySubAdd
pack_voice_SLQSVoiceDialCall_t, 345

pCallType
pack_voice_SLQSVoiceDialCall_t, 345

pCardResult
unpack_uim_ReadTransparent_t, 612
unpack_uim_SLQSUIMAuthenticate_t, 614
unpack_uim_SLQSUIMGetFileAttributes_t, 616

pCardStatus
unpack_uim_GetCardStatus_t, 611
unpack_uim_SLQSUIMSetStatusChangeCall-Back_ind_t, 618

pCcResultType
unpack_voice_SLQSOriginateUSSD_t, 620

pCellDataMask
pack_pds_ResetPDSData_t, 298

pCellDb
pack_loc_Delete_Assist_Data_t, 257

pChangeDuration
pack_nas_SLQSInitiateNetworkRegistration_t, 278

pChgDuration
pack_nas_SLQSSetSysSelectionPref_t, 296

pClientPerformOperationFlag
pack_swiavms_SLQSAVMSSendSelection_t, 321

pClkInfo
pack_loc_Delete_Assist_Data_t, 257

pColorCode
unpack_nas_SLQSNasGetHDRColorCode_t, 541

pConfig
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
_unpack_swiavms_SLQSAVMSSessionGetInfo_t, 34

pConfigAltitudeAssumed
pack_loc_Start_t, 271

pConfigurationMask
pack_uim_SLQSUIMGetConfiguration_t, 334

pConnectNumInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 629

pConnectionRequest
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33

pConnectionRetryTimers
pack_swiavms_SLQSAVMSSetSettings_t, 323
unpack_swiavms_SLQSAVMSGetSettings_t, 601

pCreateProfileOut
unpack_wds_SLQSCreateProfile_t, 658

pCurAMRConfig
unpack_voice_SLQSVoiceGetConfig_t, 639

pCurPrefVoiceSO
unpack_voice_SLQSVoiceGetConfig_t, 639

pCurProfile
pack_wds_SLQSCreateProfile_t, 370

pCurVoiceDomainPref
unpack_voice_SLQSVoiceGetConfig_t, 639

pCurVoicePrivacyPref
unpack_voice_SLQSVoiceGetConfig_t, 639

pCurrChannelRateInd
pack_wds_SLQSWdsSetEventReport_t, 382

pCurrDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 382

pCurrImgInfo
unpack_dms_SLQSSwiGetFirmwareCurr_t, 474

pCurrPrefDataSysInd
pack_wds_SLQSWdsSetEventReport_t, 382

pCurrTTYMode
unpack_voice_SLQSVoiceGetConfig_t, 639

pCurrentPersonality
unpack_nas_SLQSSwiGetHDRPersonality_t, 548

pCurrentPrsnlty
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 548

pCustSettingInfo
DMSgetCustomFeatureV2, 59

pCustSettingList
DMSgetCustomFeatureV2, 59

pCustomSCP
pack_nas_SetCDMANetworkParameters_t, 274

pDDTMInd
pack_nas_SLQSNasIndicationRegisterExt_t, 286

PDOP
loc_precisionDilution, 104

PDPType
unpack_wds_SLQSGetRuntimeSettings_t, 667

pDRCParams
unpack_nas_SLQSSwiGetHRPDStats_t, 549

pData
pack_cat_CATSendEnvelopeCommand_t, 239

pack_cat_CATSendTerminalResponse_t, 240
pDataBearer
unpack_wds_GetDataBearerTechnology_t, 647
pDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 382
pDataCallStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 382
pDataMode
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pDataRate
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pDataSessionStatus
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
pDataStatusDetail
unpack_nas_SLQSSwiNetworkDebug_t, 551
pDataSystemStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 382
pDayltSavAdj
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 543
pDefaultPDNEnabled
pack_wds_SLQSSet3GPPConfigItem_t, 376
pDeferTime
pack_swiavms_SLQSAVMSSendSelection_t, 321
pack_swioma_SLQSOMADMSendSelection_t, 327
pDestSMSContent
pack_dms_SLQSSwiSetDyingGaspCfg_t, 249
packgetDyingGaspCfg, 384
pDestSMSNum
pack_dms_SLQSSwiSetDyingGaspCfg_t, 249
packgetDyingGaspCfg, 384
pDetachAction
pack_nas_SLQSSwiPSDetach_t, 297
pDeviceConfigDetail
unpack_nas_SLQSSwiNetworkDebug_t, 551
pDiagInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pDigitBuff
voice_burstDTMFInfo, 687
pDirNum
unpack_nas_SLQSNasGet3GPP2Subscription_t, 539
pDormancyStatusInd
pack_wds_SLQSWdsSetEventReport_t, 382
pDualStandByPrefInd
pack_nas_SLQSNasIndicationRegisterExt_t, 286
pDuration
unpack_pds_GetXTRAValidity_t, 562
pECIOThresList
nas_ECIOThresh, 133
pECIOThresh
pack_nas_SLQSConfigSigInfo_t, 277
PER
nas_NetworkStatEVDO, 174
pEVDOPageMonPerChangeInd
pack_wds_SLQSWdsSetEventReport_t, 382
pEmerMode
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pEmergencyCategory
pack_voice_SLQSVoiceDialCall_t, 345
pEnableNotification
pack_wds_DHCPv4ClientLeaseChange_t, 359
pEnabled
pack_wds_SetMobileIPProfile_t, 369
pEnabledStatus
unpack_pds_GetPDSState_t, 559
pEncryptData
pack_uim_ReadTransparent_t, 330
pEncryptedData
unpack_uim_ReadTransparent_t, 612
pEncryptedPIN1
pack_uim_VerifyPin_t, 340
unpack_uim_ChangePin_t, 611
unpack_uim_SetPinProtection_t, 613
unpack_uim_UnblockPin_t, 618
unpack_uim_VerifyPin_t, 619
pErrorRateInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pExtErrCode
UnPackGetProfileSettingOut, 675
pExtErrorCode
unpack_wds_SLQSModifyProfile_t, 668
pFailCause
unpack_voice_SLQSVoiceGetCallBarring_t, 625
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 627
unpack_voice_SLQSVoiceGetCallWaiting_t, 631
unpack_voice_SLQSVoiceGetCLIP_t, 632
unpack_voice_SLQSVoiceGetCLIR_t, 633
unpack_voice_SLQSVoiceGetCNAP_t, 634
unpack_voice_SLQSVoiceGetCOLP_t, 636
unpack_voice_SLQSVoiceGetCOLR_t, 637
unpack_voice_SLQSVoiceManageCalls_t, 639
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 641
unpack_voice_SLQSVoiceSetSUPSService_t, 643
pFailureReason
unpack_wds_SLQSStartDataSession_t, 673
pFileAttributes
unpack_uim_SLQSUIMGetFileAttributes_t, 616
pFixId
unpack_loc_PositionRpt_Ind_t, 501
pFlashPayLd
pack_voice_SLQSVoiceSendFlash_t, 353
pFlashType
pack_voice_SLQSVoiceSendFlash_t, 353
pFollowOnDC
sms_sendAsyncsmsParams, 403
pForceOnDC
sms_sendAsyncsmsParams, 403
pForceRev0

pack_nas_SetCDMANetworkParameters_t, 275
pFwAutoCheck
pack_swioma_SLQSOMADMSetSettings_t, 328
pFwAutoSDM
pack_swiavms_SLQSAVMSSetSettings_t, 323
pGERANInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pGPRSMinimumQoS
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pGPRSRequestedQos
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pGPSDataMask
pack_pds_ResetPDSData_t, 298
pGPSWeek
unpack_pds_GetXTRAValidity_t, 562
pGPSWeekOffset
unpack_pds_GetXTRAValidity_t, 562
pGSMBER
unpack_nas_SLQSGetErrorRate_t, 522
pGSMCallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pGSMCipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pGSMRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pGSMRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 282
pGSMRSSIThreshList
nas_GSMRSSIThresh, 138
pGSMSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 544
pGSMSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pGWAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pGetCallFWExtInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 627
pGetCallFWInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 627
pGetCustomInput
DMSgetCustomFeatureV2, 59
pGetDyingGaspCfg
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 473
pGetDyingGaspStatistics
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 473
pGetIndicationRegInfo
unpack_sms_SLQSGetIndicationRegister_t, 595
pGetMsgWaitingInfoResp
unpack_sms_SLQSGetMessageWaiting_t, 595
pGetTransLayerInfo
unpack_sms_SLQSGetTransLayerInfo_t, 598
pGetTransNWRegInfo
unpack_sms_SLQSGetTransNWRegInfo_t, 598
pGnssData
pack_loc_Delete_Assist_Data_t, 258
pGpsTime
unpack_loc_BestAvailPos_Ind_t, 490
unpack_loc_PositionRpt_Ind_t, 501
pHA2002bis
pack_wds_SetMobileIPParameters_t, 367
pHAAuthenticator
pack_wds_SetMobileIPParameters_t, 367
pHASPI
pack_wds_SetMobileIPProfile_t, 369
pHDRECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRECIOThreshList
nas_HDRECIOThresh, 142
pHDRIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRIOThreshList
nas_HDRIOThresh, 143
pHDRNewUATIAssInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pHDRPackErrRate
unpack_nas_SLQSGetErrorRate_t, 522
pHDRRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRRSSIThreshList
nas_HDRRSSIThresh, 143
pHDRSINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRSINRThresList
nas_HDRSINRThresh, 144
pHDRSINRThresh
pack_nas_SLQSConfigSigInfo_t, 277
pack_nas_SLQSNasConfigSigInfo2_t, 282
pHDRSINRThreshList
nas_HDRSINRThreshold, 145
pHDRSessionCloseInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pHDRSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 544
pHDRSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 534

unpack_nas_SLQSSysInfoCallback_ind_t, 554
pHTTPStatus
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
pHaltSubscription
unpack_uim_SLQSUIMGetConfiguration_t, 616
pHeading
unpack_loc_BestAvailPos_Ind_t, 490
unpack_loc_PositionRpt_Ind_t, 501
pHeadingUnc
unpack_loc_BestAvailPos_Ind_t, 490
unpack_loc_PositionRpt_Ind_t, 502
pHomeSIDNID
unpack_nas_SLQSNasGet3GPP2Subscription_t, 539
pHorCirConf
unpack_loc_BestAvailPos_Ind_t, 490
pHorConfidence
unpack_loc_PositionRpt_Ind_t, 502
pHorEllpConf
unpack_loc_BestAvailPos_Ind_t, 490
pHorReliability
unpack_loc_BestAvailPos_Ind_t, 490
unpack_loc_PositionRpt_Ind_t, 502
pHorUncCircular
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pHorUncEllipseOrientAzimuth
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pHorUncEllipseSemiMajor
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pHorUncEllipseSemiMinor
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pHorizontalAccuracyLvl
pack_loc_Start_t, 272
pHorizontalConfidence
pack_pds_SLQSPDSInjectPositionData_t, 305
pHorizontalUncCircular
pack_pds_SLQSPDSInjectPositionData_t, 305
pHotSwapStatus
unpack_uim_GetCardStatus_t, 611
pHwConfig
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 379
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 671
PI
voice_callFWExtInfo, 691
pIOThresList
nas_IOThresh, 149
pIOThresh
pack_nas_SLQSConfigSigInfo_t, 277
pIPv4AddrInfo
unpack_loc_GetServer_Ind_t, 495
pIPv4AddrPref
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pIPv6AddPref
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pIPv6AddrInfo
unpack_loc_GetServer_Ind_t, 495
pIgnoreHotSwapSwitch
pack_uim_SLQSUIMPowerUp_t, 335
pImCnFlag
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pImageList
pack_fms_SetImagesPreference_t, 257
unpack_fms_GetImagesPreference_t, 485
pIndicationToken
pack_uim_ChangePin_t, 329
pack_uim_ReadTransparent_t, 330
pack_uim_SetPinProtection_t, 331
pack_uim_SLQSUIMAuthenticate_t, 332
pack_uim_SLQSUIMGetFileAttributes_t, 334
pack_uim_UnblockPin_t, 339
pack_uim_VerifyPin_t, 340
unpack_uim_ChangePin_t, 611
unpack_uim_ReadTransparent_t, 612
unpack_uim_SetPinProtection_t, 613
unpack_uim_SLQSUIMAuthenticate_t, 614
unpack_uim_SLQSUIMGetFileAttributes_t, 616
unpack_uim_UnblockPin_t, 618
unpack_uim_VerifyPin_t, 619
pInfo
pack_voice_AnswerUSSD_t, 341
pack_voice_OriginateUSSD_t, 341
pIntermediateReportState
pack_loc_Start_t, 272
pInterval
unpack_pds_GetPDSDefaults_t, 559
unpack_pds_GetXTRAAutomaticDownload_t, 561
pIpcpAckTimeout
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pIpcpCreqRetryCount
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pIsPcscfAddressNedded
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pKeyReferenceID
pack_uim_ChangePin_t, 329
pack_uim_SetPinProtection_t, 332
pack_uim_UnblockPin_t, 339
pack_uim_VerifyPin_t, 340
PLMNData
nas_operatorPLMNList, 177
PLMNName
nas_operatorNameString, 175
PLMNNetName
nas_PLMNNetworkName, 183
PLMNRecID

nas_OperatorPLMNData, 176
pLTEAttachProfile
pack_wds_SLQSSet3GPPConfigItem_t, 376
pLTEAttachProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 376
pLTEBandPref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pLTECphyCa
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pLTEInfo
unpack_nas_SLQSNasSwiModemStatus_t, 546
pLTEInfoInterfreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pLTEInfoIntrafreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pLTEInfoNeighboringGSM
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pLTEInfoNeighboringWCDMA
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pLTERSRPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 282
pLTERSRPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTERSRPThreshList
nas_LTERSRPThresh, 159
pLTERSRQDelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTERSRQThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTERSRQThreshList
nas_LTERSRQThresh, 160
pLTERSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTERSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTERSSIThreshList
nas_LTERSSIThresh, 160
pLTESNRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTESNRThresList
nas_LTESNRThresh, 163
pLTESNRThresh
pack_nas_SLQSConfigSigInfo_t, 277
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTESNRThreshList
nas_LTESNRThreshold, 164
pLTESigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 544
pLTESigRptCfg
pack_nas_SLQSConfigSigInfo_t, 277
pLTESigRptConfig
pack_nas_SLQSNasConfigSigInfo2_t, 283
pLTESrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pLTEVoiceSupportSysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pLatitude
pack_pds_SLQSPDSInjectPositionData_t, 305
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pLcpAckTimeout
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pLcpCreqRetryCount
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pLeapSeconds
unpack_loc_PositionRpt_Ind_t, 502
pLinktimer
pack_sms_SendSMS_t, 313
sms_sendAsyncsmsParams, 403
pLongitude
pack_pds_SLQSPDSInjectPositionData_t, 305
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pLteEARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 289
unpack_nas_SLQSNASSwiGetChannelLock_t, 545
pLtePCI
pack_nas_SLQSNASSwiSetChannelLock_t, 289
unpack_nas_SLQSNASSwiGetChannelLock_t, 545
pMCC
unpack_nas_GetHomeNetwork3GPP2_t, 510
pMIPStatusInd
pack_wds_SLQSWdsSetEventReport_t, 382
pMNAAA
pack_wds_SetMobileIPProfile_t, 369
pMNC
unpack_nas_GetHomeNetwork3GPP2_t, 510
pMNCIncPCSDigStat
pack_nas_SLQSSetSysSelectionPref_t, 296
pMNHA
pack_wds_SetMobileIPProfile_t, 369
pMNRInfo
pack_nas_SLQSInitiateNetworkRegistration_t, 279
pMagneticDeviation
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pManagedRoamingInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pMaxStorageSizeReq
pack_sms_SLQSSmsGetMaxStorageSize_t, 320
pMaxStorageSizeResp
unpack_sms_SLQSSmsGetMaxStorageSize_t, 599
pMessage
pack_sms_SaveSMS_t, 312
pack_sms_SendSMS_t, 313

sms_sendAsyncsmsParams, 403
pMessageIndex
pack_sms_SLQSDeleteSMS_t, 314
unpack_sms_SaveSMS_t, 593
pMessageMode
pack_sms_SLQSDeleteSMS_t, 314
pack_sms_SLQSGetSMS_t, 315
pack_sms_SLQSGetSMSList_t, 316
pack_sms_SLQSModifySMSStatus_t, 317
sms_maxStorageSizeReq, 397
pMessageProtocol
unpack_sms_SLQSSmsGetMessageProtocol_t, 599
pMessageTag
pack_sms_SLQSDeleteSMS_t, 314
pMinBasedIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 539
pMinIntervalTime
pack_loc_Start_t, 272
pMncPcsDigitStatus
pack_nas_SLQSInitiateNetworkRegistration_t, 279
pMncPcsStatus
pack_nas_SLQSGetPLMNName_t, 278
pMode
pack_wds_SetMobileIPParameters_t, 367
pModePref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pNAI
pack_wds_SetMobileIPProfile_t, 369
pNAMNameInfo
unpack_nas_SLQSNasGet3GPP2Subscription_t, 539
pNID
unpack_nas_GetHomeNetwork3GPP2_t, 510
pNITZInformation
unpack_nas_SLQSGetOperatorNameData_t, 524
pNamID
pack_voice_SLQSVoiceGetConfig_t, 350
pName
pack_wds_SetDefaultProfile_t, 365
unpack_nas_GetHomeNetwork3GPP2_t, 510
pNegoDnsSrvrPref
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pNeighborSetPilotPN
nas_NetworkStat1x, 172
pNetSelPref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pNetworkMode
pack_pds_SLQSGetAGPSConfig_t, 303
pack_pds_SLQSSetAGPSConfig_t, 306
pNetworkRejectInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pNetworkStat1x
unpack_nas_SLQSSwiNetworkDebug_t, 551
pNetworkStatEVDO
unpack_nas_SLQSSwiNetworkDebug_t, 551
pNetworkTimeInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pNotifStore
pack_swiavms_SLQSAVMSSetSettings_t, 323
pNotification
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
_unpack_swiavms_SLQSAVMSSessionGetInfo_t, 34
pNotificationStore
unpack_swiavms_SLQSAVMSGetSettings_t, 601
pNumberOfPhySlot
unpack_uim_SLQSUIMGetSlotsStatus_t, 617
pNw2DescDisp
unpack_nas_GetHomeNetwork3GPP2_t, 510
pNw2DescEnc
unpack_nas_GetHomeNetwork3GPP2_t, 510
pNw2DescLen
unpack_nas_GetHomeNetwork3GPP2_t, 510
pNw2MCC
unpack_nas_GetHomeNetwork3GPP2_t, 510
pNw2MNC
unpack_nas_GetHomeNetwork3GPP2_t, 510
pNw2Name
unpack_nas_GetHomeNetwork3GPP2_t, 510
pOTASPStatus
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pObjectVer
unpack_nas_SLQSSwiNetworkDebug_t, 551
pOperation
unpack_pds_GetPDSDefaults_t, 559
pOperatorNameString
unpack_nas_SLQSGetOperatorNameData_t, 524
pOperatorPLMNList
unpack_nas_SLQSGetOperatorNameData_t, 524
pOptList
wds_DHCPv4OptionList, 719
wdsDhcpv4OptionList, 730
pOptVal
wds_DHCPOpt, 717
pPCSInstance
unpack_nas_PerformNetworkScan_t, 518
pPCSInstanceSize
unpack_nas_PerformNetworkScan_t, 518
pPDNInactivTimeout
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPDNInactivTimeout3GPP2
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pPDPtype
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPLMNNetworkName
unpack_nas_SLQSGetOperatorNameData_t, 524

pPRLPref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pPWD
_PackSwiAVMSSettingsAPNInfo, 31
pPackageID
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
_unpack_swiavms_SLQSAVMSSessionGetInfo_t, 34
pack_swiavms_SLQSAVMSSendSelection_t, 321
pPass
pack_wds_SLQSStartDataSession_t, 380
pPassword
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pack_wds_SetDefaultProfile_t, 365
pPasswordSize
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPcscfAddrUsingDhcp
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPcscfAddrUsingPCO
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPdnType
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pPdpAccessConFlag
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPdpContext
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPdpDataCompType
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPdpHdrCompType
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPeriodInfo
pack_swiavms_SLQSAVMSSetSettings_t, 323
pPeroidsInfo
unpack_swiavms_SLQSAVMSGetSettings_t, 601
pPersonalityListLength
unpack_nas_SLQSSwiGetHDRPersonality_t, 548
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 548
pPersonalizationStatus
unpack_uim_SLQSUIMGetConfiguration_t, 616
pPilotSetData
unpack_nas_SLQSSwiGetHRPDStats_t, 549
pPilotSetInfo
nas_PilotSetData, 182
pPollingTimer
pack_swiavms_SLQSAVMSSetSettings_t, 323
unpack_swiavms_SLQSAVMSGetSettings_t, 601
pPositionSource
pack_pds_SLQSPDSInjectPositionData_t, 305
pPppSessCloseTimer1x
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pPppSessCloseTimerDO
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pPrecisionDilution
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pPrefVoiceDomain
pack_voice_SLQSVoiceSetConfig_t, 355
pPrefVoicePrivacy
pack_voice_SLQSVoiceGetConfig_t, 350
pPrefVoiceSO
pack_voice_SLQSVoiceGetConfig_t, 350
pack_voice_SLQSVoiceSetConfig_t, 355
pPrefVoiceSOStatus
unpack_voice_SLQSVoiceSetConfig_t, 642
pPreference
unpack_pds_GetXTRANetwork_t, 561
pPriDNSIPv4AddPref
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPriDNSIPv6addpref
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPriV6DnsAddress
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pPrimaryHA
pack_wds_SetMobileIPProfile_t, 369
pPrimaryID
LibpackProfile3GPP, 73
LibPackprofile_3GPP, 84
pPrimaryV4DnsAddress
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pProfileID
unpack_wds_SLQSCreateProfile_t, 658
pProfileId
pack_wds_SLQSCreateProfile_t, 370
pack_wds_SLQSModifyProfile_t, 375
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 378
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 379
pProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 376
pProfileSettings
unpack_wds_SLQSGetProfileSettings_t, 665
pProfileType
pack_wds_SLQSCreateProfile_t, 370
pack_wds_SLQSModifyProfile_t, 375
pProfilename
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pProfilenameSize
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85

pProtoSubTypElmnt
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 548
pProtocol
pack_nas_SetCDMANetworkParameters_t, 275
pProtocolSubtypeElement
unpack_nas_SLQSSwiGetHDRPersonality_t, 548
pQosClassID
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pRAT
pack_nas_SLQSSetSysSelectionPref_t, 296
pRATINstance
unpack_nas_PerformNetworkScan_t, 518
pRATInstanceSize
unpack_nas_PerformNetworkScan_t, 518
pRATType
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
PRIString
unpack_dms_GetFirmwareRevision_t, 451
unpack_dms_GetFirmwareRevisions_t, 452
PRLInd
unpack_nas_SLQSGetServingSystem_t, 529
PRLPTlv
NASQmiCbkNasSystemSelPrefInd, 235
PRLPref
NASPRLPreferenceTlv, 233
pRSRPThresList
nas_RSRPThresh, 193
pRSRPThresh
pack_nas_SLQSConfigSigInfo_t, 277
pRSRQThresList
nas_RSRQThresh, 194
pRSRQThresh
pack_nas_SLQSConfigSigInfo_t, 277
pRSSIThresList
nas_RSSIThresh, 194
pRSSIThresh
pack_nas_SLQSConfigSigInfo_t, 277
pRXChain0Info
unpack_nas_SLQSNasGetTxRxInfo_t, 542
pRXChain1Info
unpack_nas_SLQSNasGetTxRxInfo_t, 542
pRXDroppedCount
unpack_wds_GetPacketStatistics_t, 654
pRXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 654
pRXOkBytesCount
unpack_wds_GetPacketStatistics_t, 654
pRXPacketErrors
unpack_wds_GetPacketStatistics_t, 654
pRXPacketOverflows
unpack_wds_GetPacketStatistics_t, 654
pRXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 654
pRXTotalBytes
unpack_wds_GetByteTotals_t, 646
pRadioInterface
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 543
pRankIndicatorInd
pack_nas_SLQSNasSwiIndicationRegister_t, 288
pReRegPeriod
pack_wds_SetMobileIPParameters_t, 367
pReRegTraffic
pack_wds_SetMobileIPParameters_t, 367
pReadResult
unpack_uim_ReadTransparent_t, 612
pReasonMask
unpack_dms_GetOfflineReason_t, 456
pRecurrenceType
pack_loc_Start_t, 272
pRefreshEvent
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 617
pRegCallStatInfoEvt
sms_getIndicationReg, 395
sms_setIndicationReg, 404
pRegDTMFEvents
pack_voice_SLQSVoiceIndicationRegister_t, 351
pRegInd
sms_getTransLayerInfo, 396
pRegStatus
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
sms_getTransNWRegInfo, 396
pRegTransLayerInfoEvt
sms_getIndicationReg, 395
sms_setIndicationReg, 404
pRegTransNWRegInfoEvt
sms_getIndicationReg, 395
sms_setIndicationReg, 404
pRegVoicePrivacyEvents
pack_voice_SLQSVoiceIndicationRegister_t, 351
pRejectReason
pack_swiavms_SLQSAVMSSendSelection_t, 321
pack_swioma_SLQSOMADMSendSelection_t, 327
pRemainingRetries
unpack_uim_ChangePin_t, 611
unpack_uim_SetPinProtection_t, 613
unpack_uim_SLQSUIMDepersonalization_t, 615
unpack_uim_UnblockPin_t, 618
unpack_uim_VerifyPin_t, 619
pRemotePartyName
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pRemotePartyNum
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pReportChannelRate
pack_wds_SLQSGetDUNCallInfo_t, 372
pReportConnStatus
pack_wds_SLQSGetDUNCallInfo_t, 372
pReportDataBearerTech
pack_wds_SLQSGetDUNCallInfo_t, 372
pReportDormStatus
pack_wds_SLQSGetDUNCallInfo_t, 372

pReqSettings
pack_wds_SLQSGetRuntimeSettings_t, 374
pRequestOptionList
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 379
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 671
pRequestedTag
pack_sms_SLQSGetSMSList_t, 316
pRetryInterval
pack_wds_SetMobileIPParameters_t, 367
pRetryLimit
pack_wds_SetMobileIPParameters_t, 367
pRetryMessage
sms_sendAsyncsmsParams, 403
pRetryMessageId
sms_sendAsyncsmsParams, 403
pRevInUse
nas_CDMASysInfo, 121
pRevInUseValid
nas_CDMASysInfo, 121
pRevTunneling
pack_wds_SetMobileIPProfile_t, 369
pRoamPref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pRoamTimer
pack_voice_SLQSVoiceGetConfig_t, 350
pRoamTimerCnt
unpack_voice_SLQSVoiceGetConfig_t, 639
pRoamTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 355
pRoamTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 642
pRoaming
pack_nas_SetCDMANetworkParameters_t, 275
pRscp
unpack_nas_SLQSNasSigInfoCallback_ind_t, 544
PSDomain
unpack_nas_GetServingNetwork_t, 515
pSID
unpack_nas_GetHomeNetwork3GPP2_t, 510
pSMSAttemptedFlag
packgetDyingGaspStatistics, 385
pSMSCAddress
pack_sms_SetSMSCAddress_t, 313
unpack_sms_GetSMSCAddress_t, 592
pSMSCType
pack_sms_SetSMSCAddress_t, 313
unpack_sms_GetSMSCAddress_t, 593
pSPC
pack_nas_SetCDMANetworkParameters_t, 275
pack_wds_SetMobileIPParameters_t, 367
pSV
loc_BdsSVInfo, 97
loc_SVInfo, 108
pSVInfo
pack_loc_Delete_Assist_Data_t, 258
pSarRFState
unpack_sar_SLQSGetRfSarState_t, 592
pSatelliteInfo
unpack_loc_GnssSvInfo_Ind_t, 496
pScanResult
unpack_nas_PerformNetworkScan_t, 518
pSccRxInfo
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 550
pSecDNSIPv4AddPref
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pSecDNSIPv6addpref
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pSecV6DnsAddress
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pSecondaryFlag
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pSecondaryHA
pack_wds_SetMobileIPProfile_t, 369
pSecondaryV4DnsAddress
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pSectorID
nas_NetworkStatEVDO, 174
pSendSmsParams
pack_sms_SLQSSendAsyncSMS_t, 317
pSensorDataUsage
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pServerAddress
pack_pds_SLQSSetAGPSConfig_t, 306
unpack_pds_SLQSGetAGPSConfig_t, 565
pServerPort
pack_pds_SLQSSetAGPSConfig_t, 306
unpack_pds_SLQSGetAGPSConfig_t, 565
pServerURL
pack_pds_SLQSSetAGPSConfig_t, 306
unpack_pds_SLQSGetAGPSConfig_t, 565
pServerURLLength
pack_pds_SLQSSetAGPSConfig_t, 306
unpack_pds_SLQSGetAGPSConfig_t, 565
pServiceClass
pack_voice_SLQSVoiceSetSUPSService_t, 358
pServiceOption
sms_sendAsyncsmsParams, 403
pServingSystemInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pSessionType
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
pSetIndicationRegReq
pack_sms_SLQSSetIndicationRegister_t, 318
pSetRoutesReq
pack_sms_SLQSSmsSetRoutes_t, 320
pSigIndReq

pack_nas_SLQSSetSignalStrengthsCallback_t, 291
pSignalStrengthInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pSmsOnIms
sms_sendAsyncsmsParams, 403
pSmsStorage
unpack_sms_SLQSSwiGetSMSStorage_t, 600
pSourceIP
LibPackTFTIDParams, 93
pSpeedHorizontal
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pSpeedUnc
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pSpeedVertical
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pSpeedVerticalUnc
unpack_loc_BestAvailPos_Ind_t, 491
pSrvDomainPref
pack_nas_SLQSSetSysSelectionPref_t, 296
unpack_nas_SLQSGetSysSelectionPref_t, 538
pSrvOpt
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pSrvRegRestriction
pack_nas_SLQSSetSysSelectionPref_t, 296
pSrvcProviderName
unpack_nas_SLQSGetOperatorNameData_t, 524
pStatMask
pack_wds_GetPacketStatistics_t, 362
pSubscriptionInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pSuppressSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pSuppsNotifEvents
pack_voice_SLQSVoiceIndicationRegister_t, 351
pSvUsedforFix
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pSvcClass
pack_voice_SLQSVoiceGetCallBarring_t, 347
pack_voice_SLQSVoiceGetCallForwardingStatus_t, 347
pack_voice_SLQSVoiceGetCallWaiting_t, 348
unpack_voice_SLQSVoiceGetCallBarring_t, 625
unpack_voice_SLQSVoiceGetCallWaiting_t, 631
pSvcType
pack_voice_SLQSVoiceDialCall_t, 345
pSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pSysInfoNoChange
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pSystemSelectionInd
pack_nas_SLQSNasIndicationRegisterExt_t, 287
pTDSCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMAECIOThreshList
nas_TDSCDMAECIOThresh, 208
pTDSCDMARSCPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMARSCPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMARSCPThreshList
nas_TDSCDMARSCPThresh, 208
pTDSCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMARSSIThreshList
nas_TDSCDMARSSIThresh, 209
pTDSCDMASINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMASINRThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pTDSCDMASINRThreshList
nas_TDSCDMASINRThresh, 210
pTDSCDMASigInfoExt
unpack_nas_SLQSNasSigInfoCallback_ind_t, 544
pTFTID1Params
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pTFTID2Params
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pTTYConfigStatus
unpack_voice_SLQSVoiceSetConfig_t, 642
pTTYMode
pack_voice_SLQSVoiceGetConfig_t, 350
pack_voice_SLQSVoiceSetConfig_t, 355
pTXDroppedCount
unpack_wds_GetPacketStatistics_t, 654
pTXInfo
unpack_nas_SLQSNasGetTxRxInfo_t, 542
pTXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 654
pTXOkBytesCount
unpack_wds_GetPacketStatistics_t, 654
pTXPacketErrors
unpack_wds_GetPacketStatistics_t, 654
pTXPacketOverflows
unpack_wds_GetPacketStatistics_t, 654
pTXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 654
pTXTotalBytes
unpack_wds_GetByteTotals_t, 646
pTdscdmaBandPref
pack_nas_SLQSSetSysSelectionPref_t, 297
pTech
pack_wds_SLQSStartDataSession_t, 380
pTechnologyMask
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502

pTimeSrc
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pTimeStamp
pack_pds_SLQSPDSInjectPositionData_t, 305
packgetDyingGaspStatistics, 385
pTimeType
pack_pds_SLQSPDSInjectPositionData_t, 305
pTimeUnc
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pTimeZone
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 543
pTimeout
unpack_pds_GetPDSDefaults_t, 559
pTimer
pack_nas_SLQSNasSwiIndicationRegister_t, 288
pTimerVal
pack_voice_SLQSVoiceSetSUPSService_t, 358
pTimestampUtc
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pTrackingStatus
unpack_pds_GetPDSState_t, 559
pTransLayerInfo
sms_getTransLayerInfo, 396
pTransferStatInd
pack_wds_SLQSGetDUNCallInfo_t, 372
pack_wds_SLQSWdsSetEventReport_t, 382
pTransferStatusReport
sms_setRoutesReq, 405
pTrueIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 539
pUATI
unpack_nas_SLQSSwiGetHRPDStats_t, 549
pUMTSCellID
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pUMTSInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pUMTSMinQoS
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pUMTSMinQosSigInd
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pUMTSReqQoS
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pUMTSReqQoSSigInd
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pURL
unpack_loc_GetServer_Ind_t, 495
pUSSDInfo
unpack_voice_SLQSOriginateUSSD_t, 620
pUUSInFo
pack_voice_SLQSVoiceDialCall_t, 345
pUUSInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pUimSlotsStatus
unpack_uim_SLQSUIMGetSlotsStatus_t, 617
pUname
_PackSwiAVMSSettingsAPNInfo, 31
pUser
pack_wds_SLQSStartDataSession_t, 380
pUserData
sms_sendAsyncsmsParams, 403
pUserId
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pUserIdSize
LibpackProfile3GPP2, 79
LibPackprofile_3GPP2, 90
pUsername
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pack_wds_SetDefaultProfile_t, 365
pUsernameSize
LibpackProfile3GPP, 74
LibPackprofile_3GPP, 85
pVerboseFailReasonType
unpack_wds_SLQSStartDataSession_t, 673
pVerboseFailureReason
unpack_wds_SLQSStartDataSession_t, 673
pVertConfidence
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pVertReliability
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pVertUnc
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_PositionRpt_Ind_t, 502
pVerticalConfidence
pack_pds_SLQSPDSInjectPositionData_t, 305
pVerticalUnc
pack_pds_SLQSPDSInjectPositionData_t, 305
pVoiceDomainPref
pack_voice_SLQSVoiceGetConfig_t, 350
pVoiceDomainPrefStatus
unpack_voice_SLQSVoiceSetConfig_t, 642
pVoicePrivacy
unpack_voice_SLQSVoiceGetAllCallInfo_t, 624
unpack_voice_SLQSVoiceGetCallInfo_t, 629
pWAMSParaChanged
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
pWCDMABER
unpack_nas_SLQSGetErrorRate_t, 522
pWCDMACallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pWCDMACipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 534

unpack_nas_SLQSSysInfoCallback_ind_t, 554
pWCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pWCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pWCDMAECIOThreshList
nas_WCDMAECIOThresh, 219
pWCDMAInfoLTENeighborCell
unpack_nas_SLQSNasGetCellLocationInfo_t, 540
pWCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 283
pWCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 283
pWCDMARSSIThreshList
nas_WCDMARSSIThresh, 221
pWCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 544
pWCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 534
unpack_nas_SLQSSysInfoCallback_ind_t, 554
pWcdmaUARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 289
unpack_nas_SLQSNASSwiGetChannelLock_t, 545
pWifiState
pack_pds_SLQSSetPositionMethodState_t, 307
pXid
unpack_loc_BestAvailPos_Ind_t, 491
pXtraDataState
pack_pds_SLQSSetPositionMethodState_t, 307
pXtraTimeState
pack_pds_SLQSSetPositionMethodState_t, 307
pack_cat_CATSendEnvelopeCommand
cat.h, 732
pack_cat_CATSendEnvelopeCommand_t, 238
cmdID, 239
dataLen, 239
pData, 239
pack_cat_CATSendTerminalResponse
cat.h, 732
pack_cat_CATSendTerminalResponse_t, 239
dataLen, 240
pData, 240
refID, 240
pack_cat_SetCATEventCallback
cat.h, 733
pack_cat_SetCATEventCallback_t, 240
eventMask, 240
pack_dms_ActivateAutomatic
dms.h, 747
pack_dms_ActivateAutomatic_t, 241
actCode, 241
pack_dms_GetActivationState
dms.h, 747
pack_dms_GetBandCapability
dms.h, 747
pack_dms_GetCrashAction
dms.h, 748
pack_dms_GetCustFeature
dms.h, 748
pack_dms_GetCustFeaturesV2
dms.h, 749
pack_dms_GetCustFeaturesV2_t, 241
cust_id, 241
list_type, 241
Tlvresult, 241
pack_dms_GetDeviceCap
dms.h, 749
pack_dms_GetDeviceCapabilities
dms.h, 749
pack_dms_GetDeviceHardwareRev
dms.h, 750
pack_dms_GetDeviceMfr
dms.h, 750
pack_dms_GetDeviceSerialNumbers
dms.h, 750
pack_dms_GetFSN
dms.h, 752
pack_dms_GetFirmwareInfo
dms.h, 751
pack_dms_GetFirmwareRevision
dms.h, 751
pack_dms_GetFirmwareRevisions
dms.h, 752
pack_dms_GetHardwareRevision
dms.h, 752
pack_dms_GetIMSI
dms.h, 753
pack_dms_GetManufacturer
dms.h, 753
pack_dms_GetModelID
dms.h, 753
pack_dms_GetNetworkTime
dms.h, 754
pack_dms_GetOfflineReason
dms.h, 754
pack_dms_GetPRLVersion
dms.h, 755
pack_dms_GetPower
dms.h, 754
pack_dms_GetSerialNumbers
dms.h, 755
pack_dms_GetUSBComp
dms.h, 755
pack_dms_GetVoiceNumber
dms.h, 756
pack_dms_ResetToFactoryDefaults
dms.h, 756
pack_dms_ResetToFactoryDefaults_t, 242
spc, 242
pack_dms_SLQSDmsSwiGetResetInfo
dms.h, 760
pack_dms_SLQSDmsSwiIndicationRegister

dms.h, 760
pack_dms_SLQSDmsSwiIndicationRegister_t, 248
resetInfoInd, 248
pack_dms_SLQSGetBandCapability
dms.h, 760
pack_dms_SLQSGetERIFile
dms.h, 761
pack_dms_SLQSSwiClearDyingGaspStatistics
dms.h, 761
pack_dms_SLQSSwiGetCrashInfo
dms.h, 761
pack_dms_SLQSSwiGetCrashInfo_t, 248
clear, 248
pack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 762
pack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 762
pack_dms_SLQSSwiGetFirmwareCurr
dms.h, 762
pack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 763
pack_dms_SLQSSwiGetHostDevInfo
dms.h, 763
pack_dms_SLQSSwiGetOSInfo
dms.h, 763
pack_dms_SLQSSwiGetSerialNoExt
dms.h, 764
pack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 764
pack_dms_SLQSSwiSetDyingGaspCfg_t, 249
pDestSMSContent, 249
pDestSMSNum, 249
pack_dms_SLQSSwiSetHostDevInfo
dms.h, 764
pack_dms_SLQSSwiSetHostDevInfo_t, 249
manString, 250
modelString, 250
plasmaIDString, 250
swVerString, 250
pack_dms_SLQSSwiSetOSInfo
dms.h, 765
pack_dms_SLQSSwiSetOSInfo_t, 250
nameString, 250
versionString, 250
pack_dms_SLQSUIMGetState
dms.h, 765
pack_dms_SetActivationStatusCallback
dms.h, 757
pack_dms_SetActivationStatusCallback_t, 242
activationState, 242
pack_dms_SetCrashAction
dms.h, 757
pack_dms_SetCrashAction_t, 242
crashAction, 243
pack_dms_SetCustFeature
dms.h, 757
pack_dms_SetCustFeature_t, 243
DHCPRelayEnabled, 245
DisableIMSI, 245
GPSLPM, 245
GPSSel, 245
GpsEnable, 245
IPFamSupport, 245
IsVoiceEnabled, 245
RMAutoConnect, 245
SMSSupport, 245
pack_dms_SetCustFeaturesV2
dms.h, 758
pack_dms_SetCustFeaturesV2_t, 245
cust_id, 246
cust_value, 246
Tlvresult, 246
value_length, 246
pack_dms_SetEventReport
dms.h, 758
pack_dms_SetEventReport_t, 246
mode, 246
pack_dms_SetFirmwarePreference
dms.h, 758
pack_dms_SetPower
dms.h, 759
pack_dms_SetPower_t, 246
mode, 247
Tlvresult, 247
pack_dms_SetUSBComp
dms.h, 759
pack_dms_SetUSBComp_t, 247
Tlvresult, 248
USBComp, 248
pack_dms_UIMChangePIN
dms.h, 765
pack_dms_UIMChangePIN_t, 250
id, 251
newValue, 251
oldValue, 251
pack_dms_UIMGetControlKeyStatus
dms.h, 766
pack_dms_UIMGetControlKeyStatus_t, 251
facility, 251
pack_dms_UIMGetICCID
dms.h, 766
pack_dms_UIMGetICCID_t, 252
Tlvresult, 252
pack_dms_UIMGetPINStatus
dms.h, 767
pack_dms_UIMSetControlKeyProtection
dms.h, 767
pack_dms_UIMSetControlKeyProtection_t, 252
facility, 253
facilityCk, 253
facilityState, 253
pack_dms_UIMSetPINProtection
dms.h, 767
pack_dms_UIMSetPINProtection_t, 253
bEnable, 253
id, 253

value, 253
pack_dms_UIMUnblockControlKey
dms.h, 768
pack_dms_UIMUnblockControlKey_t, 253
facility, 254
facilityCk, 254
pack_dms_UIMUnblockPIN
dms.h, 768
pack_dms_UIMUnblockPIN_t, 254
id, 254
newPin, 254
pukValue, 254
pack_dms_UIMVerifyPIN
dms.h, 768
pack_dms_UIMVerifyPIN_t, 255
id, 255
value, 255
pack_dms_ValidateSPC
dms.h, 769
pack_fms_GetImagesPreference
fms.h, 793
pack_fms_GetImagesPreference_t, 255
Tlvresult, 255
pack_fms_GetStoredImages
fms.h, 794
pack_fms_GetStoredImages_t, 255
Tlvresult, 256
pack_fms_SetImagesPreference
fms.h, 794
pack_fms_SetImagesPreference_t, 256
bForceDownload, 257
imageListSize, 257
modemindex, 257
pImageList, 257
Tlvresult, 257
pack_loc_Delete_Assist_Data_t, 257
pBdsSVInfo, 257
pCellDb, 257
pClkInfo, 257
pGnssData, 258
pSVInfo, 258
Tlvresult, 258
pack_loc_DeleteAssistData
loc.h, 808
pack_loc_EventRegister
loc.h, 808
pack_loc_EventRegister_t, 258
eventRegister, 260
Tlvresult, 260
pack_loc_SLQSLOCGetBestAvailPos
loc.h, 810
pack_loc_SLQSLOCGetBestAvailPos_t, 261
Tlvresult, 261
xid, 261
pack_loc_SLQSLOCGetServer
loc.h, 810
pack_loc_SLQSLOCGetServer_t, 262
has_serverAddrTypeMask, 262
serverAddrTypeMask, 262
serverType, 262
pack_loc_SLQSLOCInjectPosition
loc.h, 810
pack_loc_SLQSLOCInjectPosition_t, 262
altitudeSrcInfo, 266
altitudeWrtEllipsoid, 266
altitudeWrtMeanSeaLevel, 266
has_altitudeSrcInfo, 266
has_altitudeWrtEllipsoid, 266
has_horConfidence, 266
has_horReliability, 266
has_horUncCircular, 266
has_latitude, 266
has_longitude, 266
has_positionSrc, 266
has_rawHorConfidence, 266
has_rawHorUncCircular, 266
has_timestampAge, 266
has_timestampUtc, 266
has_vertConfidence, 266
has_vertRelicability, 266
has_vertUnc, 267
horConfidence, 267
horReliability, 267
horUncCircular, 267
latitude, 267
longitude, 267
positionSrc, 267
rawHorConfidence, 267
rawHorUncCircular, 267
timestampAge, 267
timestampUtc, 267
vertConfidence, 267
vertReliability, 267
vertUnc, 267
pack_loc_SLQSLOCInjectSensorData
loc.h, 811
pack_loc_SLQSLOCInjectSensorData_t, 267
accelTemp, 268
acceleroData, 268
acceleroTimeSrc, 268
gyroData, 268
gyroTemp, 268
gyroTimeSrc, 269
has_accelTemp, 269
has_acceleroTimeSrc, 269
has_accleroData, 269
has_gyroData, 269
has_gyroTemp, 269
has_gyroTimeSrc, 269
has_opaqueId, 269
opaqueId, 269
pack_loc_SLQSLOCInjectUTCTime
loc.h, 811
pack_loc_SLQSLOCInjectUTCTime_t, 269
timeMsec, 269
timeUncMsec, 269

pack_loc_SLQSLOCSetCradleMountConfig
loc.h, 811
pack_loc_SLQSLOCSetCradleMountConfig_t, 269
confidence, 270
has_confidence, 270
state, 270
pack_loc_SetExtPowerState
loc.h, 809
pack_loc_SetExtPowerState_t, 260
extPowerState, 260
Tlvresult, 260
pack_loc_SetOperationMode
loc.h, 809
pack_loc_SetOperationMode_t, 260
mode, 261
Tlvresult, 261
pack_loc_Start
loc.h, 812
pack_loc_Start_t, 270
pApplicationInfo, 271
pConfigAltitudeAssumed, 271
pHorizontalAccuracyLvl, 272
pIntermediateReportState, 272
pMinIntervalTime, 272
pRecurrenceType, 272
SessionId, 272
Tlvresult, 272
pack_loc_Stop
loc.h, 812
pack_loc_Stop_t, 272
SessionId, 272
Tlvresult, 272
pack_nas_GetACCOLC
nas.h, 828
pack_nas_GetANAAAAuthenticationStatus
nas.h, 829
pack_nas_GetCDMANetworkParameters
nas.h, 829
pack_nas_GetHomeNetwork
nas.h, 829
pack_nas_GetHomeNetwork3GPP2
nas.h, 830
pack_nas_GetNetworkPreference
nas.h, 830
pack_nas_GetRFInfo
nas.h, 830
pack_nas_GetServingNetwork
nas.h, 830
pack_nas_GetServingNetworkCapabilities
nas.h, 831
pack_nas_GetSignalStrengths
nas.h, 831
pack_nas_InitiateDomainAttach
nas.h, 831
pack_nas_InitiateDomainAttach_t, 272
action, 273
pack_nas_PerformNetworkScan
nas.h, 832
pack_nas_SLQSConfigSigInfo
nas.h, 834
pack_nas_SLQSConfigSigInfo_t, 276
pECIOThresh, 277
pIOThresh, 277
pLTESNRThresh, 277
pLTESigRptCfg, 277
pRSRPThresh, 277
pRSRQThresh, 277
pRSSIThresh, 277
pack_nas_SLQSGetErrorRate
nas.h, 834
pack_nas_SLQSGetNetworkTime
nas.h, 834
pack_nas_SLQSGetOperatorNameData
nas.h, 835
pack_nas_SLQSGetPLMNName
nas.h, 835
pack_nas_SLQSGetPLMNName_t, 277
mcc, 278
mnc, 278
pMncPcsStatus, 278
pack_nas_SLQSGetServingSystem
nas.h, 835
pack_nas_SLQSGetSignalStrength
nas.h, 836
pack_nas_SLQSGetSysInfo
nas.h, 836
pack_nas_SLQSGetSysSelectionPref
nas.h, 836
pack_nas_SLQSInitiateNetworkRegistration
nas.h, 837
pack_nas_SLQSInitiateNetworkRegistration_t, 278
pChangeDuration, 278
pMNRInfo, 279
pMncPcsDigitStatus, 279
regAction, 279
pack_nas_SLQSNASSwiGetChannelLock
nas.h, 840
pack_nas_SLQSNASSwiSetChannelLock
nas.h, 841
pack_nas_SLQSNASSwiSetChannelLock_t, 288
pack_nas_SLQSNasConfigSigInfo2
nas.h, 837
pack_nas_SLQSNasConfigSigInfo2_t, 279
pHDRIODelta, 282
pHDRIOThresh, 282
pLTESigRptConfig, 283
pack_nas_SLQSNasGet3GPP2Subscription
nas.h, 837
pack_nas_SLQSNasGet3GPP2Subscription_t, 283
namID, 284
pack_nas_SLQSNasGetCellLocationInfo
nas.h, 838
pack_nas_SLQSNasGetHDRColorCode
nas.h, 838
pack_nas_SLQSNasGetSigInfo
nas.h, 838

pack_nas_SLQSNasGetTxRxInfo
nas.h, 839
pack_nas_SLQSNasGetTxRxInfo_t, 284
radio_if, 284
pack_nas_SLQSNasIndicationRegisterExt
nas.h, 839
pack_nas_SLQSNasIndicationRegisterExt_t, 284
pDDTMInd, 286
pDualStandByPrefInd, 286
pErrorRateInd, 287
pHDRSessionCloseInd, 287
pLTECphyCa, 287
pManagedRoamingInd, 287
pNetworkRejectInd, 287
pNetworkTimeInd, 287
pServingSystemInd, 287
pSignalStrengthInd, 287
pSubscriptionInfoInd, 287
pSuppressSysInfoInd, 287
pSysInfoInd, 287
pSystemSelectionInd, 287
pack_nas_SLQSNasSwiIndicationRegister
nas.h, 840
pack_nas_SLQSNasSwiIndicationRegister_t, 287
gsmUmtsDl, 288
gsmUmtsUl, 288
lteEmmDl, 288
lteEmmUl, 288
lteEsmDl, 288
lteEsmUl, 288
pRankIndicatorInd, 288
pTimer, 288
pack_nas_SLQSNasSwiModemStatus
nas.h, 840
pack_nas_SLQSSetBandPreference
nas.h, 841
pack_nas_SLQSSetBandPreference_t, 289
bandPref, 290
pack_nas_SLQSSetSignalStrengthsCallback
nas.h, 841
pack_nas_SLQSSetSignalStrengthsCallback_t, 291
bEnable, 291
pSigIndReq, 291
pack_nas_SLQSSetSysSelectionPref
nas.h, 842
pack_nas_SLQSSetSysSelectionPref_t, 291
pAcqOrderPref, 296
pBandPref, 296
pCSGID, 296
pChgDuration, 296
pEmerMode, 296
pGWAcqOrderPref, 296
pLTEBandPref, 296
pModePref, 296
pNetSelPref, 296
pPRLPref, 296
pRAT, 296
pRoamPref, 296
pSrvDomainPref, 296
pSrvRegRestriction, 296
pTdscdmaBandPref, 297
pack_nas_SLQSSwiGetHDRPersonality
nas.h, 842
pack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 842
pack_nas_SLQSSwiGetHRPDStats
nas.h, 843
pack_nas_SLQSSwiGetLteCQI
nas.h, 843
pack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 843
pack_nas_SLQSSwiNetworkDebug
nas.h, 844
pack_nas_SLQSSwiPSDetach
nas.h, 844
pack_nas_SLQSSwiPSDetach_t, 297
pDetachAction, 297
pack_nas_SetACCOLC
nas.h, 832
pack_nas_SetACCOLC_t, 273
accolc, 273
spc, 273
pack_nas_SetCDMANetworkParameters
nas.h, 832
pack_nas_SetCDMANetworkParameters_t, 273
pApplication, 274
pBroadcast, 274
pCustomSCP, 274
pForceRev0, 275
pProtocol, 275
pRoaming, 275
pSPC, 275
pack_nas_SetLURejectCallback
nas.h, 833
pack_nas_SetNetworkPreference
nas.h, 833
pack_nas_SetNetworkPreference_t, 275
Duration, 275
TechnologyPref, 275
Tlvresult, 275
pack_nas_SetRFInfoCallback
nas.h, 833
pack_nas_SlqsGetLTECphyCAInfo
nas.h, 834
pack_pds_ForceXTRADownload
pds.h, 866
pack_pds_GetPDSDefaults
pds.h, 866
pack_pds_GetPDSState
pds.h, 866
pack_pds_GetPortAutomaticTracking
pds.h, 867
pack_pds_GetServiceAutomaticTracking
pds.h, 867
pack_pds_GetXTRAAutomaticDownload
pds.h, 867

pack_pds_GetXTRANetwork
pds.h, 868
pack_pds_GetXTRAValidity
pds.h, 868
pack_pds_PDSInjectTimeReference
pds.h, 868
pack_pds_PDSInjectTimeReference_t, 297
systemDiscontinuities, 297
systemTime, 297
pack_pds_ResetPDSData
pds.h, 869
pack_pds_ResetPDSData_t, 298
pCellDataMask, 298
pGPSDataMask, 298
pack_pds_SLQSGetAGPSConfig
pds.h, 872
pack_pds_SLQSGetAGPSConfig_t, 302
pNetworkMode, 303
pack_pds_SLQSGetGPSStateInfo
pds.h, 872
pack_pds_SLQSPDSDeterminePosition
pds.h, 873
pack_pds_SLQSPDSInjectAbsoluteTimeReference
pds.h, 873
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 303
forceFlag, 303
timeBase, 303
timeMsec, 303
timeUncMsec, 303
pack_pds_SLQSPDSInjectPositionData
pds.h, 873
pack_pds_SLQSPDSInjectPositionData_t, 303
pAltitudeWrtEllipsoid, 305
pAltitudeWrtSealevel, 305
pHorizontalConfidence, 305
pHorizontalUncCircular, 305
pLatitude, 305
pLongitude, 305
pPositionSource, 305
pTimeStamp, 305
pTimeType, 305
pVerticalConfidence, 305
pVerticalUnc, 305
pack_pds_SLQSSetAGPSConfig
pds.h, 874
pack_pds_SLQSSetAGPSConfig_t, 305
pNetworkMode, 306
pServerAddress, 306
pServerPort, 306
pServerURL, 306
pack_pds_SLQSSetPositionMethodState
pds.h, 874
pack_pds_SLQSSetPositionMethodState_t, 306
pWifiState, 307
pXtraDataState, 307
pXtraTimeState, 307
pack_pds_SetEventReportCallback
pds.h, 869
pack_pds_SetEventReportCallback_t, 298
posDataNmea, 299
rptPosData, 299
pack_pds_SetPDSDefaults
pds.h, 870
pack_pds_SetPDSDefaults_t, 299
accuracy, 300
interval, 300
operation, 300
timeout, 300
pack_pds_SetPDSState
pds.h, 870
pack_pds_SetPDSState_t, 300
enable, 300
pack_pds_SetPortAutomaticTracking
pds.h, 870
pack_pds_SetPortAutomaticTracking_t, 300
bAuto, 301
pack_pds_SetServiceAutomaticTracking
pds.h, 871
pack_pds_SetServiceAutomaticTracking_t, 301
bAuto, 301
pack_pds_SetXTRAAutomaticDownload
pds.h, 871
pack_pds_SetXTRAAutomaticDownload_t, 301
bEnabled, 302
interval, 302
pack_pds_SetXTRANetwork
pds.h, 871
pack_pds_SetXTRANetwork_t, 302
preference, 302
pack_pds_StartPDSTrackingSessionExt
pds.h, 874
pack_pds_StartPDSTrackingSessionExt_t, 307
fixAccuracy, 308
fixCount, 308
fixInterval, 308
fixTimeout, 308
sessionControl, 308
sessionOperation, 308
sessionServerOption, 308
sessionType, 308
pack_pds_StopPDSTrackingSession
pds.h, 875
pack_qmi_t, 308
msgid, 309
svc, 309
timeout, 309
xid, 309
pack_qos_SLQSQosGetNetworkStatus
qos.h, 933
pack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 933
pack_qos_SLQSQosSwiReadApnExtraParams_t, 309
apnId, 309
pack_qos_SLQSQosSwiReadDataStats
qos.h, 934

pack_qos_SLQSQosSwiReadDataStats_t, 309
apnId, 310
pack_qos_SLQSSetQosEventCallback
qos.h, 935
pack_qos_SLQSSetQosEventCallback_t, 310
enable, 310
pack_rms_GetSMSWake
rms.h, 940
pack_rms_SetSMSWake
rms.h, 940
pack_rms_SetSMSWake_t, 310
enabled, 310
wake_mask, 311
pack_sar_SLQSGetRfSarState
sar.h, 941
pack_sar_SLQSSetRfSarState
sar.h, 942
pack_sar_SLQSSetRfSarState_t, 311
RfSarState, 311
pack_sms_GetSMSCAddress
sms.h, 948
pack_sms_SLQSDeleteSMS
sms.h, 950
pack_sms_SLQSDeleteSMS_t, 314
pMessageIndex, 314
pMessageMode, 314
pMessageTag, 314
storageType, 314
pack_sms_SLQSGetIndicationRegister
sms.h, 950
pack_sms_SLQSGetMessageWaiting
sms.h, 951
pack_sms_SLQSGetSMS
sms.h, 951
pack_sms_SLQSGetSMS_t, 314
messageIndex, 315
pMessageMode, 315
storageType, 315
pack_sms_SLQSGetSMSList
sms.h, 952
pack_sms_SLQSGetSMSList_t, 315
pMessageMode, 316
pRequestedTag, 316
storageType, 316
pack_sms_SLQSGetSmsBroadcastConfig
sms.h, 951
pack_sms_SLQSGetSmsBroadcastConfig_t, 315
mode, 315
pack_sms_SLQSGetTransLayerInfo
sms.h, 952
pack_sms_SLQSGetTransNWRegInfo
sms.h, 953
pack_sms_SLQSModifySMSStatus
sms.h, 953
pack_sms_SLQSModifySMSStatus_t, 316
messageIndex, 317
messageTag, 317
pMessageMode, 317
storageType, 317
pack_sms_SLQSSendAsyncSMS
sms.h, 953
pack_sms_SLQSSendAsyncSMS_t, 317
pSendSmsParams, 317
pack_sms_SLQSSetIndicationRegister
sms.h, 954
pack_sms_SLQSSetIndicationRegister_t, 317
pSetIndicationRegReq, 318
pack_sms_SLQSSetSmsBroadcastActivation
sms.h, 954
pack_sms_SLQSSetSmsBroadcastActivation_t, 318
broadcastActivate, 318
mode, 318
pack_sms_SLQSSetSmsBroadcastConfig
sms.h, 954
pack_sms_SLQSSetSmsBroadcastConfig_t, 318
mode, 319
pBroadcastConfig, 319
pCDMABroadcastConfig, 319
pack_sms_SLQSSetSmsStorage
sms.h, 955
pack_sms_SLQSSetSmsStorage_t, 319
smsStorage, 319
pack_sms_SLQSSmsGetMaxStorageSize
sms.h, 955
pack_sms_SLQSSmsGetMaxStorageSize_t, 319
pMaxStorageSizeReq, 320
pack_sms_SLQSSmsGetMessageProtocol
sms.h, 955
pack_sms_SLQSSmsSetRoutes
sms.h, 956
pack_sms_SLQSSmsSetRoutes_t, 320
pSetRoutesReq, 320
pack_sms_SLQSSwiGetSMSStorage
sms.h, 956
pack_sms_SaveSMS
sms.h, 948
pack_sms_SaveSMS_t, 311
messageFormat, 312
messageSize, 312
pMessage, 312
storageType, 312
pack_sms_SendSMS
sms.h, 949
pack_sms_SendSMS_t, 312
messageFormat, 313
messageSize, 313
pLinktimer, 313
pMessage, 313
pack_sms_SetNewSMSCallback
sms.h, 949
pack_sms_SetNewSMSCallback_t, 313
status, 313
pack_sms_SetSMSCAddress
sms.h, 950
pack_sms_SetSMSCAddress_t, 313
pSMSCAddress, 313

pSMSCType, 313
pack_swiavms_SLQSAVMSGetSettings
swiavms.h, 974
pack_swiavms_SLQSAVMSSendSelection
swiavms.h, 975
pack_swiavms_SLQSAVMSSendSelection_t, 320
pDeferTime, 321
pPackageID, 321
pRejectReason, 321
selection, 321
pack_swiavms_SLQSAVMSSessionGetInfo
swiavms.h, 975
pack_swiavms_SLQSAVMSSetSettings
swiavms.h, 976
pack_swiavms_SLQSAVMSSetSettings_t, 321
AutoConnect, 322
AutoReboot, 322
FwAutoUpdate, 322
FwAutodownload, 322
pAPNInfo, 323
pConnectionRetryTimers, 323
pNotifStore, 323
pPeriodInfo, 323
pPollingTimer, 323
pack_swiavms_SLQSAVMSSetSettingsNoAutoReboot-
Field
swiavms.h, 977
pack_swiavms_SLQSAVMSStartSession
swiavms.h, 978
pack_swiavms_SLQSAVMSStartSession_t, 323
sessionType, 323
pack_swiavms_SLQSAVMSStopSession
swiavms.h, 978
pack_swiavms_SLQSAVMSStopSession_t, 323
sessionType, 324
pack_swiavms_SLQSAvmsSetEventReport
swiavms.h, 976
pack_swiloc_SwiLocGetAutoStart
swiloc.h, 984
pack_swiloc_SwiLocSetAutoStart
swiloc.h, 985
pack_swiloc_SwiLocSetAutoStart_t, 324
fix_rate, 325
fix_type, 325
function, 325
max_dist, 325
max_time, 325
set_fix_rate, 325
set_fix_type, 325
set_function, 325
set_max_dist, 325
set_max_time, 325
pack_swioma_SLQSOMADMAlertCallback
swioma.h, 987
pack_swioma_SLQSOMADMCancelSession
swioma.h, 987
pack_swioma_SLQSOMADMCancelSession_t, 325
sessionType, 326
pack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 988
pack_swioma_SLQSOMADMGetSessionInfo_t, 326
SessionType, 326
pack_swioma_SLQSOMADMGetSettings
swioma.h, 989
pack_swioma_SLQSOMADMSendSelection
swioma.h, 989
pack_swioma_SLQSOMADMSendSelection_t, 326
pDeferTime, 327
pRejectReason, 327
selection, 327
pack_swioma_SLQSOMADMSetSettings
swioma.h, 990
pack_swioma_SLQSOMADMSetSettings_t, 327
FOTAdownload, 328
pAutosdm, 328
pFwAutoCheck, 328
pack_swioma_SLQSOMADMStartSession
swioma.h, 991
pack_swioma_SLQSOMADMStartSession_t, 328
sessionType, 328
pack_uim_ChangePin
uim.h, 998
pack_uim_ChangePin_t, 329
changePIN, 329
EncryptedPIN1, 329
pIndicationToken, 329
pKeyReferenceID, 329
sessionInfo, 329
Tlvresult, 329
pack_uim_GetCardStatus
uim.h, 998
pack_uim_ReadTransparent
uim.h, 999
pack_uim_ReadTransparent_t, 330
fileIndex, 330
pEncryptData, 330
pIndicationToken, 330
readTransparent, 330
sessionInfo, 331
Tlvresult, 331
pack_uim_SLQSUIMAuthenticate
uim.h, 999
pack_uim_SLQSUIMAuthenticate_t, 332
authData, 332
pIndicationToken, 332
sessionInfo, 332
pack_uim_SLQSUIMDepersonalization
uim.h, 1000
pack_uim_SLQSUIMDepersonalization_t, 332
depersonilisationInfo, 333
pack_uim_SLQSUIMEventRegister
uim.h, 1000
pack_uim_SLQSUIMEventRegister_t, 333
eventMask, 333
pack_uim_SLQSUIMGetConfiguration
uim.h, 1000

pack_uim_SLQSUIMGetConfiguration_t, 333
pConfigurationMask, 334
pack_uim_SLQSUIMGetFileAttributes
uim.h, 1001
pack_uim_SLQSUIMGetFileAttributes_t, 334
fileIndex, 334
pIndicationToken, 334
sessionInfo, 334
pack_uim_SLQSUIMGetSlotsStatus
uim.h, 1001
pack_uim_SLQSUIMPowerDown
uim.h, 1002
pack_uim_SLQSUIMPowerDown_t, 334
slot, 335
pack_uim_SLQSUIMPowerUp
uim.h, 1002
pack_uim_SLQSUIMPowerUp_t, 335
pIgnoreHotSwapSwitch, 335
slot, 335
pack_uim_SLQSUIMRefreshComplete
uim.h, 1002
pack_uim_SLQSUIMRefreshComplete_t, 335
refreshComplete, 336
sessionInfo, 336
pack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 1003
pack_uim_SLQSUIMRefreshGetLastEvent_t, 336
sessionInfo, 336
pack_uim_SLQSUIMRefreshOK
uim.h, 1003
pack_uim_SLQSUIMRefreshOK_t, 336
OKtoRefresh, 337
sessionInfo, 337
pack_uim_SLQSUIMRefreshRegister
uim.h, 1003
pack_uim_SLQSUIMRefreshRegister_t, 337
regRefresh, 337
sessionInfo, 337
pack_uim_SLQSUIMReset
uim.h, 1004
pack_uim_SLQSUIMSwitchSlot
uim.h, 1004
pack_uim_SLQSUIMSwitchSlot_t, 337
bLogicalSlot, 338
ulPhysicalSlot, 338
pack_uim_SetPinProtection
uim.h, 999
pack_uim_SetPinProtection_t, 331
EncryptedPIN1, 331
pIndicationToken, 331
pKeyReferenceID, 332
pinProtection, 331
sessionInfo, 332
Tlvresult, 332
pack_uim_UnblockPin
uim.h, 1004
pack_uim_UnblockPin_t, 338
EncryptedPIN1, 339
pIndicationToken, 339
pKeyReferenceID, 339
pinProtection, 339
sessionInfo, 339
Tlvresult, 339
pack_uim_VerifyPin
uim.h, 1005
pack_uim_VerifyPin_t, 339
pEncryptedPIN1, 340
pIndicationToken, 340
pKeyReferenceID, 340
sessionInfo, 340
Tlvresult, 340
verifyPIN, 340
pack_voice_AnswerUSSD
voice.h, 1018
pack_voice_AnswerUSSD_t, 340
pInfo, 341
pack_voice_CancelUSSD
voice.h, 1018
pack_voice_OriginateUSSD
voice.h, 1018
pack_voice_OriginateUSSD_t, 341
pInfo, 341
pack_voice_SLQSOriginateUSSD
voice.h, 1019
pack_voice_SLQSOriginateUSSD_t, 341
ussDCS, 342
ussData, 341
ussLen, 342
pack_voice_SLQSVoiceALSSelectLine
voice.h, 1019
pack_voice_SLQSVoiceALSSelectLine_t, 342
lineValue, 342
pack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 1019
pack_voice_SLQSVoiceALSSetLineSwitching_t, 342
switchOption, 342
pack_voice_SLQSVoiceAnswerCall
voice.h, 1020
pack_voice_SLQSVoiceAnswerCall_t, 342
pCallId, 343
pack_voice_SLQSVoiceBindSubscription
voice.h, 1020
pack_voice_SLQSVoiceBindSubscription_t, 343
subsType, 343
pack_voice_SLQSVoiceBurstDTMF
voice.h, 1021
pack_voice_SLQSVoiceBurstDTMF_t, 343
pack_voice_SLQSVoiceDialCall
voice.h, 1021
pack_voice_SLQSVoiceDialCall_t, 344
callNumber, 345
pCLIRType, 345
pCUGInfo, 345
pCallPartySubAdd, 345
pCallType, 345
pEmergencyCategory, 345

pSvcType, 345
pUUSInFo, 345
pack_voice_SLQSVoiceEndCall
voice.h, 1021
pack_voice_SLQSVoiceEndCall_t, 346
pCallId, 346
pack_voice_SLQSVoiceGetAllCallInfo
voice.h, 1022
pack_voice_SLQSVoiceGetCLIP
voice.h, 1024
pack_voice_SLQSVoiceGetCLIR
voice.h, 1024
pack_voice_SLQSVoiceGetCNAP
voice.h, 1024
pack_voice_SLQSVoiceGetCOLP
voice.h, 1025
pack_voice_SLQSVoiceGetCOLR
voice.h, 1025
pack_voice_SLQSVoiceGetCallBarring
voice.h, 1022
pack_voice_SLQSVoiceGetCallBarring_t, 346
pSvcClass, 347
reason, 347
pack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 1022
pack_voice_SLQSVoiceGetCallForwardingStatus_t, 347
pSvcClass, 347
Reason, 347
pack_voice_SLQSVoiceGetCallInfo
voice.h, 1023
pack_voice_SLQSVoiceGetCallInfo_t, 347
callID, 348
pack_voice_SLQSVoiceGetCallWaiting
voice.h, 1023
pack_voice_SLQSVoiceGetCallWaiting_t, 348
pSvcClass, 348
pack_voice_SLQSVoiceGetConfig
voice.h, 1025
pack_voice_SLQSVoiceGetConfig_t, 348
pAMRStatus, 350
pAirTimer, 350
pAutoAnswer, 350
pNamID, 350
pPrefVoicePrivacy, 350
pPrefVoiceSO, 350
pRoamTimer, 350
pTTYMode, 350
pVoiceDomainPref, 350
pack_voice_SLQSVoiceIndicationRegister
voice.h, 1026
pack_voice_SLQSVoiceIndicationRegister_t, 350
pRegDTMFEvents, 351
pRegVoicePrivacyEvents, 351
pSuppsNotifEvents, 351
pack_voice_SLQSVoiceManageCalls
voice.h, 1026
pack_voice_SLQSVoiceManageCalls_t, 351
pCallID, 351
SUPSType, 352
pack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 1026
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 352
pack_voice_SLQSVoiceSendFlash
voice.h, 1027
pack_voice_SLQSVoiceSendFlash_t, 352
pCallID, 353
pFlashPayLd, 353
pFlashType, 353
pack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 1027
pack_voice_SLQSVoiceSetCallBarringPassword_t, 353
newPasswd, 354
newPasswdAgain, 354
oldPasswd, 354
Reason, 354
pack_voice_SLQSVoiceSetConfig
voice.h, 1027
pack_voice_SLQSVoiceSetConfig_t, 354
pAirTimerConfig, 355
pAutoAnswer, 355
pPrefVoiceDomain, 355
pPrefVoiceSO, 355
pRoamTimerConfig, 355
pTTYMode, 355
pack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 1028
pack_voice_SLQSVoiceSetPreferredPrivacy_t, 355
privacyPref, 355
pack_voice_SLQSVoiceSetSUPSService
voice.h, 1028
pack_voice_SLQSVoiceSetSUPSService_t, 355
pCallBarringPasswd, 358
pCallForwardingNumber, 358
pServiceClass, 358
pTimerVal, 358
reason, 358
voiceSvc, 358
pack_voice_SLQSVoiceStartContDTMF
voice.h, 1029
pack_voice_SLQSVoiceStartContDTMF_t, 358
DTMFdigit, 358
pCallID, 358
pack_voice_SLQSVoiceStopContDTMF
voice.h, 1029
pack_voice_SLQSVoiceStopContDTMF_t, 359
callID, 359
pack_wds_DHCPv4ClientLeaseChange
wds.h, 1046
pack_wds_DHCPv4ClientLeaseChange_t, 359
pEnableNotification, 359
pack_wds_GetAutoconnect
wds.h, 1047
pack_wds_GetByteTotals
wds.h, 1047
pack_wds_GetConnectionRate
wds.h, 1048

pack_wds_GetDataBearerTechnology
wds.h, 1048
pack_wds_GetDefaultProfile
wds.h, 1048
pack_wds_GetDefaultProfile_t, 359
profiletype, 360
pack_wds_GetDefaultProfileNum
wds.h, 1049
pack_wds_GetDefaultProfileNum_t, 360
family, 360
type, 360
pack_wds_GetDormancyState
wds.h, 1049
pack_wds_GetDormancyState_t, 360
pack_wds_GetLastMobileIPError
wds.h, 1050
pack_wds_GetLastMobileIPError_t, 360
pack_wds_GetMobileIP
wds.h, 1050
pack_wds_GetMobileIP_t, 361
pack_wds_GetMobileIPProfile
wds.h, 1050
pack_wds_GetMobileIPProfile_t, 361
index, 361
pack_wds_GetPacketStatistics
wds.h, 1051
pack_wds_GetPacketStatistics_t, 361
pStatMask, 362
pack_wds_GetPacketStatus
wds.h, 1051
pack_wds_GetPacketStatus_t, 362
statmask, 362
pack_wds_GetSessionDuration
wds.h, 1052
pack_wds_GetSessionDuration_t, 362
pack_wds_GetSessionState
wds.h, 1052
pack_wds_RMSetTransferStatistics
wds.h, 1053
pack_wds_RMSetTransferStatistics_t, 362
RmTrasnferStaticsReq, 363
pack_wds_SLQSCreateProfile
wds.h, 1055
pack_wds_SLQSCreateProfile_t, 369
pCurProfile, 370
pProfileId, 370
pProfileType, 370
pack_wds_SLQSDeleteProfile
wds.h, 1056
pack_wds_SLQSDeleteProfile_t, 370
profileIndex, 370
profileType, 370
pack_wds_SLQSGet3GPPConfigItem
wds.h, 1056
pack_wds_SLQSGetCurrDataSystemStat
wds.h, 1057
pack_wds_SLQSGetCurrDataSystemStat_t, 370
pack_wds_SLQSGetCurrentChannelRate
wds.h, 1057
pack_wds_SLQSGetDUNCallInfo
wds.h, 1058
pack_wds_SLQSGetDUNCallInfo_t, 371
Mask, 372
pReportChannelRate, 372
pReportConnStatus, 372
pReportDataBearerTech, 372
pReportDormStatus, 372
pTransferStatInd, 372
pack_wds_SLQSGetDataBearerTechnology
wds.h, 1057
pack_wds_SLQSGetDataBearerTechnology_t, 371
pack_wds_SLQSGetProfileSettings
wds.h, 1058
pack_wds_SLQSGetProfileSettings_t, 372
ProfileId, 373
ProfileType, 373
pack_wds_SLQSGetRuntimeSettings
wds.h, 1059
pack_wds_SLQSGetRuntimeSettings_t, 373
pReqSettings, 374
pack_wds_SLQSModifyProfile
wds.h, 1059
pack_wds_SLQSModifyProfile_t, 374
curProfile, 375
pProfileId, 375
pProfileType, 375
pack_wds_SLQSResetPacketStatics
wds.h, 1060
pack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 1061
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 378
pProfileId, 378
pack_wds_SLQSSGetLoopback
wds.h, 1062
pack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 1062
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 378
pHwConfig, 379
pProfileId, 379
pack_wds_SLQSSSetLoopback
wds.h, 1062
pack_wds_SLQSSSetLoopback_t, 379
loopbackMode, 379
loopbackMultiplier, 379
pack_wds_SLQSSet3GPPConfigItem
wds.h, 1060
pack_wds_SLQSSet3GPPConfigItem_t, 375
p3gppRelease, 376
pProfileList, 376
pack_wds_SLQSSetIPFamilyPreference
wds.h, 1060
pack_wds_SLQSSetIPFamilyPreference_t, 377
IPFamilyPreference, 377
pack_wds_SLQSSetWdsEventCallback
wds.h, 1061
pack_wds_SLQSSetWdsEventCallback_t, 377

currentDataBearer, 377
dataBearer, 378
dataBearerTechExt, 378
dataSystemStatus, 378
dormancyStatus, 378
interval, 378
mobileIP, 378
transferStats, 378
pack_wds_SLQSStartDataSession
wds.h, 1063
pack_wds_SLQSStartDataSession_t, 379
pAuth, 380
pPass, 380
pTech, 380
pUser, 380
pprofileid3gpp, 380
pprofileid3gpp2, 380
pack_wds_SLQSStopDataSession
wds.h, 1063
pack_wds_SLQSStopDataSession_t, 380
psid, 381
pack_wds_SLQSWdsGoActive
wds.h, 1064
pack_wds_SLQSWdsGoDormant
wds.h, 1064
pack_wds_SLQSWdsSetEventReport
wds.h, 1064
pack_wds_SLQSWdsSetEventReport_t, 381
pCurrChannelRateInd, 382
pCurrDataBearerTechInd, 382
pCurrPrefDataSysInd, 382
pDataBearerTechInd, 382
pDataCallStatusChangeInd, 382
pDataSystemStatusChangeInd, 382
pDormancyStatusInd, 382
pMIPStatusInd, 382
pTransferStatInd, 382
pack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 1065
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 383
contextId, 383
contextType, 383
pack_wds_SetAutoconnect
wds.h, 1053
pack_wds_SetAutoconnect_t, 363
acroamsetting, 363
acsetting, 363
pack_wds_SetDefaultProfile
wds.h, 1053
pack_wds_SetDefaultProfile_t, 363
authentication, 365
ipAddress, 365
pApnname, 365
pName, 365
pPassword, 365
pUsername, 365
pdpType, 365
primaryDNS, 365
profileType, 365
secondaryDNS, 365
pack_wds_SetDefaultProfileNum
wds.h, 1054
pack_wds_SetDefaultProfileNum_t, 365
family, 366
index, 366
type, 366
pack_wds_SetMobileIP
wds.h, 1054
pack_wds_SetMobileIP_t, 366
mode, 366
pack_wds_SetMobileIPParameters
wds.h, 1054
pack_wds_SetMobileIPParameters_t, 366
pHA2002bis, 367
pHAAuthenticator, 367
pMode, 367
pReRegPeriod, 367
pReRegTraffic, 367
pRetryInterval, 367
pRetryLimit, 367
pSPC, 367
pack_wds_SetMobileIPProfile
wds.h, 1055
pack_wds_SetMobileIPProfile_t, 367
index, 369
pAAASPI, 369
pAddress, 369
pEnabled, 369
pHASPI, 369
pMNAAA, 369
pMNHA, 369
pNAI, 369
pPrimaryHA, 369
pRevTunneling, 369
pSecondaryHA, 369
spc, 369
PackCreateProfileOut, 383
ExtErrorCode, 384
ProfileIndex, 384
ProfileType, 384
PackSwiAVMSSettingsAPNInfo
swiavms.h, 968
PackSwiAVMSSettingsConnectionRetryTimers
swiavms.h, 968
PackSwiAVMSSettingsPeriodsInfo
swiavms.h, 968
PackSwiAvmsSetSettingsAPNInfo
swiavms.h, 968
PackSwiAvmsSetSettingsConnectionRetryTimers
swiavms.h, 968
PackSwiAvmsSetSettingsPeriodInfo
swiavms.h, 968
package_name
unpack_omaDmFotaTlv_t, 557
packageid_str
unpack_dms_GetFirmwareInfo_t, 450

packetZone
nas_CDMASysInfo, 121
packetZoneValid
nas_CDMASysInfo, 121
packgetDyingGaspCfg, 384
pDestSMSContent, 384
pDestSMSNum, 384
packgetDyingGaspStatistics, 384
pSMSAttemptedFlag, 385
pTimeStamp, 385
path
uim_fileInfo, 426
pathLen
uim_fileInfo, 426
pbAuto
unpack_pds_GetPortAutomaticTracking_t, 560
unpack_pds_GetServiceAutomaticTracking_t, 560
pbEnabled
unpack_pds_GetXTRAAutomaticDownload_t, 561
pbPlatform
unpack_dms_GetOfflineReason_t, 456
pci
nas_cellParams, 123
nas_ltePCI, 158
nas_PhyCaAggPcellInfo, 178
nas_PhyCaAggScellIndType, 179
nas_PhyCaAggScellInfo, 182
nas_umtsLTENbrCell, 216
NASPhyCaAggPcellInfo, 230
NASPhyCaAggScellIndType, 232
NASPhyCaAggScellInfo, 233
pcsfFQDNAddress
wds_PCSCFFQDNAddressList, 724
pdpType
pack_wds_SetDefaultProfile_t, 365
pdptype
unpack_wds_GetDefaultProfile_t, 649
pds.h, 864
pack_pds_ForceXTRADownload, 866
pack_pds_GetPDSDefaults, 866
pack_pds_GetPDSState, 866
pack_pds_GetPortAutomaticTracking, 867
pack_pds_GetServiceAutomaticTracking, 867
pack_pds_GetXTRAAutomaticDownload, 867
pack_pds_GetXTRANetwork, 868
pack_pds_GetXTRAValidity, 868
pack_pds_PDSInjectTimeReference, 868
pack_pds_ResetPDSData, 869
pack_pds_SLQSGetAGPSConfig, 872
pack_pds_SLQSGetGPSStateInfo, 872
pack_pds_SLQSPDSDeterminePosition, 873
pack_pds_SLQSPDSInjectAbsoluteTimeReference, 873
pack_pds_SLQSPDSInjectPositionData, 873
pack_pds_SLQSSetAGPSConfig, 874
pack_pds_SLQSSetPositionMethodState, 874
pack_pds_SetEventReportCallback, 869
pack_pds_SetPDSDefaults, 870
pack_pds_SetPDSState, 870
pack_pds_SetPortAutomaticTracking, 870
pack_pds_SetServiceAutomaticTracking, 871
pack_pds_SetXTRAAutomaticDownload, 871
pack_pds_SetXTRANetwork, 871
pack_pds_StartPDSTrackingSessionExt, 874
pack_pds_StopPDSTrackingSession, 875
unpack_pds_ForceXTRADownload, 875
unpack_pds_GetPDSDefaults, 875
unpack_pds_GetPDSState, 876
unpack_pds_GetPortAutomaticTracking, 876
unpack_pds_GetServiceAutomaticTracking, 876
unpack_pds_GetXTRAAutomaticDownload, 877
unpack_pds_GetXTRANetwork, 877
unpack_pds_GetXTRAValidity, 877
unpack_pds_PDSInjectTimeReference, 878
unpack_pds_ResetPDSData, 878
unpack_pds_SLQSGetAGPSConfig, 881
unpack_pds_SLQSGetGPSStateInfo, 882
unpack_pds_SLQSPDSDeterminePosition, 882
unpack_pds_SLQSPDSInjectAbsoluteTime-Reference, 882
unpack_pds_SLQSPDSInjectPositionData, 883
unpack_pds_SLQSSetAGPSConfig, 883
unpack_pds_SLQSSetPositionMethodState, 883
unpack_pds_SetEventReport_Ind, 878
unpack_pds_SetEventReportCallback, 879
unpack_pds_SetPDSDefaults, 879
unpack_pds_SetPDSState, 879
unpack_pds_SetPdsState_Ind, 880
unpack_pds_SetPortAutomaticTracking, 880
unpack_pds_SetServiceAutomaticTracking, 880
unpack_pds_SetXTRAAutomaticDownload, 881
unpack_pds_SetXTRANetwork, 881
unpack_pds_StartPDSTrackingSessionExt, 884
unpack_pds_StopPDSTrackingSession, 884
peakRate
unpack_qos_tokenBucket_t, 588
peakThroughputClass
LibPackGPRSRequestedQoS, 68
wds_GPRSQoS, 721
PersistentTechPref
unpack_nas_GetNetworkPreference_t, 512
persoFeature
appStats, 46
uim_appStatus, 416
persoRetries
appStats, 46
uim_appStatus, 416
persoState
appStats, 46
uim_appStatus, 416
persoUnblockRetries
appStats, 46
uim_appStatus, 416
pfailureCause
unpack_voice_SLQSOriginateUSSD_t, 620
phase

nas_rxInfo, 195
PhyCaAggPcellInfo
NasGetLTECphyCaInfo, 226
PhyCaAggScellDlBw
NasGetLTECphyCaInfo, 226
PhyCaAggScellIndType
NasGetLTECphyCaInfo, 226
PhyCaAggScellIndex
NasGetLTECphyCaInfo, 226
PhyCaAggScellInfo
NasGetLTECphyCaInfo, 226
PhysicalLayer
nas_protocolSubtypeElement, 187
PilotEnergy
nas_NetworkStatEVDO, 174
PilotPN
nas_PilotSetParams, 183
PilotStrength
nas_PilotSetParams, 183
PilotType
nas_PilotSetParams, 183
pin1Len
uim_encryptedPIN1, 422
pin1Retries
appStats, 46
uim_appStatus, 416
pin1State
appStats, 46
uim_appStatus, 416
pin1Val
uim_encryptedPIN1, 422
pin2Retries
appStats, 46
uim_appStatus, 416
pin2State
appStats, 46
uim_appStatus, 416
pinID
uim_changeUIMPIN, 420
uim_setPINProtection, 433
uim_unblockUIMPIN, 436
uim_verifyUIMPIN, 437
pinLen
uim_changeUIMPIN, 420
uim_verifyUIMPIN, 437
pinLength
uim_setPINProtection, 433
pinOperation
uim_setPINProtection, 433
pinProtection
pack_uim_SetPinProtection_t, 331
pack_uim_UnblockPin_t, 339
pinVal
uim_changeUIMPIN, 420
uim_verifyUIMPIN, 437
pinValue
uim_setPINProtection, 433
PkgDescLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 607
PkgDescription
unpack_swioma_SLQSOMADMGetSessionInfo_t, 607
PkgName
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
PkgNameLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
pkgver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 474
PktErrRate
unpack_qos_swiQosFlow_t, 588
plasmaIDString
pack_dms_SLQSSwiSetHostDevInfo_t, 250
unpack_dms_SLQSSwiGetHostDevInfo_t, 476
plmn
nas_GERANInfo, 136
nas_LTEInfoIntrafreq, 156
nas_UMTSInfo, 215
port
loc_IPv4Info, 102
loc_IPv6Info, 102
unpack_qos_Port_t, 572
posDataNmea
pack_pds_SetEventReportCallback_t, 299
posSrc
unpack_pds_SetEventReport_Ind_t, 564
PositionDataNMEA
unpack_pds_SetEventReport_Ind_t, 564
positionSrc
pack_loc_SLQSLOCInjectPosition_t, 267
pprofileid3gpp
pack_wds_SLQSStartDataSession_t, 380
pprofileid3gpp2
pack_wds_SLQSStartDataSession_t, 380
prDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 675
prDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 675
prPCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 675
prPCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings-_t, 675
Precedence
unpack_qos_swiQosFilter_t, 583
precedenceClass
LibPackGPRSRequestedQoS, 68
wds_GPRSQoS, 721
prefNetwork
unpack_wds_SLQSGetCurrDataSystemStat_t, 660

unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
preference
pack_pds_SetXTRANetwork_t, 302
Preferred
nas_QmiNas3GppNetworkInfo, 189
prefixLen
unpack_qos_IPv6Addr_t, 571
presentationInd
voice_remotePartyNum, 710
priChA
nas_CDMAChannel, 115
priChB
nas_CDMAChannel, 115
priSize
unpack_dms_GetFirmwareRevisions_t, 452
pridns
unpack_wds_GetDefaultProfile_t, 649
pridnsv6
unpack_wds_GetDefaultProfile_t, 649
primaryDNS
pack_wds_SetDefaultProfile_t, 365
PrimaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 667
PrimaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 667
primaryHA
unpack_wds_GetMobileIPProfile_t, 653
privacyPref
pack_voice_SLQSVoiceSetPreferredPrivacy_t, 355
priver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 474
priversion_str
unpack_dms_GetFirmwareInfo_t, 450
ProfileID
unpack_wds_SLQSGetRuntimeSettings_t, 667
ProfileId
pack_wds_SLQSGetProfileSettings_t, 373
profileId
wds_DHCPProfileIdTlv, 717
wds_DHCPv4ProfileId, 719
wdsDhcpv4ProfileId, 730
ProfileId3GPP2
unpack_qos_swiQosFlow_t, 588
ProfileIdTlv
unpack_wds_DHCPv4ClientLease_ind_t, 645
ProfileIndex
PackCreateProfileOut, 384
profileIndex
pack_wds_SLQSDeleteProfile_t, 370
wds_ProfileIdentifier, 725
profileList
unpack_wds_SLQSGet3GPPConfigItem_t, 660
ProfileName
unpack_wds_SLQSGetRuntimeSettings_t, 667
ProfileType
pack_wds_SLQSGetProfileSettings_t, 373
PackCreateProfileOut, 384
unpack_wds_SLQSGetProfileSettings_t, 665
profileType
pack_wds_SetDefaultProfile_t, 365
pack_wds_SLQSDeleteProfile_t, 370
wds_DHCPProfileIdTlv, 717
wds_DHCPv4ProfileId, 719
wds_ProfileIdentifier, 725
wdsDhcpv4ProfileId, 730
profiletype
pack_wds_GetDefaultProfile_t, 360
Protocol
unpack_nas_GetCDMANetworkParameters_t, 508
ProvisionStatus
voice_CLIPResp, 695
voice_CLIRResp, 696
voice_CNAPResp, 696
voice_COLPResp, 697
voice_COLRResp, 698
psAttachState
nas_servSystem, 200
NASServingSystemInfo, 237
psBarStatus
nas_CallBarringSysInfo, 114
nas_callBarStatus, 115
psState
nas_CommInfo, 124
psc
nas_UMTSInfo, 215
nas_wcdmaCellInfo, 219
nas_WCDMASysInfo, 224
pscValid
nas_WCDMASysInfo, 224
pscsfIPv4Addr
wds_PCSCFIPv4ServerAddressList, 724
psetting
unpack_wds_GetAutoconnect_t, 645
psid
pack_wds_SLQSStopDataSession_t, 381
unpack_wds_SLQSStartDataSession_t, 673
puk1Retries
appStats, 46
uim_appStatus, 416
puk2Retries
appStats, 46
uim_appStatus, 416
pukLen
uim_unblockUIMPIN, 436
pukVal
uim_unblockUIMPIN, 437
pukValue
pack_dms_UIMUnblockPIN_t, 254

QMI_LITE_WDS_CURRENT_CALL_DB_MASK
wds.h, 1046
QMI_LITE_WDS_LAST_CALL_DB_MASK
wds.h, 1046
QCI
LibPackQosClassID, 91

QFlowState
unpack_qos_QosFlowInfo_t, 574
QLIC
nas_DeviceConfigDetail, 131
QMI pack/unpack (lite-qmi), 25
qaGobiApiTableBandClasses.h, 884
qaGobiApiTableCallControlReturnReasons.h, 887
qaGobiApiTableCallEndReasons.h, 888
qaGobiApiTableCarrierCodes.h, 904
qaGobiApiTableCodingScheme.h, 906
qaGobiApiTableGpsCapabilityCodes.h, 909
qaGobiApiTablePowerModes.h, 910
qaGobiApiTableRadioInterfaces.h, 910
qaGobiApiTableRegionCodes.h, 911
qaGobiApiTableServiceOptions.h, 911
qaGobiApiTableSupServiceInfoClasses.h, 914
qaGobiApiTableSwiAudio.h, 914
qaGobiApiTableSwiOMADMSessionStatus.h, 915
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h, 916
qaGobiApiTableVoiceCallEndReasons.h, 917
qm_wds_ds_profile_extended_err_codes
qmerrno.h, 930
qmerrno.h
eQCWWAN_ERR_API_MUTEX_TIMEOUT, 927
eQCWWAN_ERR_BUFFER_SZ, 926
eQCWWAN_ERR_CANCEL_OP, 927
eQCWWAN_ERR_DRIVER, 927
eQCWWAN_ERR_ENUM_BEGIN, 926
eQCWWAN_ERR_ENUM_END, 927
eQCWWAN_ERR_FILE_COPY, 926
eQCWWAN_ERR_FILE_OPEN, 926
eQCWWAN_ERR_GENERAL, 926
eQCWWAN_ERR_INTERNAL, 926
eQCWWAN_ERR_INVALID_ARG, 926
eQCWWAN_ERR_INVALID_DEVID, 926
eQCWWAN_ERR_INVALID_FILE, 926
eQCWWAN_ERR_INVALID_QMI_RSP, 926
eQCWWAN_ERR_INVALID_XID, 927
eQCWWAN_ERR_MALFORMED_QMI_RSP, 926
eQCWWAN_ERR_MEMORY, 926
eQCWWAN_ERR_MULTIPLE_DEVICES, 927
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED, 927
eQCWWAN_ERR_NO_CANCELABLE_OP, 927
eQCWWAN_ERR_NO_CONNECTION, 926
eQCWWAN_ERR_NO_DEVICE, 926
eQCWWAN_ERR_NO_SIGNAL, 927
eQCWWAN_ERR_NONE, 926
eQCWWAN_ERR_NULL_TLV, 930
eQCWWAN_ERR_OFFLINE, 927
eQCWWAN_ERR_PDU_GENERATION, 927
eQCWWAN_ERR_QMI_ABORTED, 927
eQCWWAN_ERR_QMI_ACCESS_DENIED, 929
eQCWWAN_ERR_QMI_ACK_NOT_SENT, 929
eQCWWAN_ERR_QMI_ARG_TOO_LONG, 927
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED, 928
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK, 928
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED, 929
eQCWWAN_ERR_QMI_CALL_FAILED, 927
eQCWWAN_ERR_QMI_CARD_BUSY_RSP, 930
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED, 929
eQCWWAN_ERR_QMI_CAT_END, 930
eQCWWAN_ERR_QMI_CAT_START, 930
eQCWWAN_ERR_QMI_CAUSE_CODE, 928
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED, 927
eQCWWAN_ERR_QMI_CONNECT, 926
eQCWWAN_ERR_QMI_DEVICE_IN_USE, 927
eQCWWAN_ERR_QMI_DEVICE_NOT_READY, 928
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL, 928
eQCWWAN_ERR_QMI_DISABLED, 928
eQCWWAN_ERR_QMI_ENCODING, 928
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE, 930
eQCWWAN_ERR_QMI_EVENT_REG_FAILED, 930
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL, 929
eQCWWAN_ERR_QMI_FDN_RESTRICT, 929
eQCWWAN_ERR_QMI_FLOW_SUSPENDED, 928
eQCWWAN_ERR_QMI_GENERAL, 928
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED, 929
eQCWWAN_ERR_QMI_IFACE, 926
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE, 929
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER, 928
eQCWWAN_ERR_QMI_INCORRECT_PIN, 927
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE, 929
eQCWWAN_ERR_QMI_INJECT_TIMEOUT, 929
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES, 928
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND, 928
eQCWWAN_ERR_QMI_INTERNAL, 927
eQCWWAN_ERR_QMI_INVALID_ARG, 928
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID, 927
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT, 928
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD, 930
eQCWWAN_ERR_QMI_INVALID_HANDLE, 927
eQCWWAN_ERR_QMI_INVALID_ID, 928
eQCWWAN_ERR_QMI_INVALID_INDEX, 928

eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF, 928
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE, 928
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID, 928
eQCWWAN_ERR_QMI_INVALID_OPERATION, 929
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE, 927
eQCWWAN_ERR_QMI_INVALID_PINID, 927
eQCWWAN_ERR_QMI_INVALID_PROFILE, 927
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE, 927
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION, 928
eQCWWAN_ERR_QMI_INVALID_QMI_CMD, 929
eQCWWAN_ERR_QMI_INVALID_QOS_ID, 928
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION, 928
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE, 927
eQCWWAN_ERR_QMI_INVALID_TECH_PREF, 927
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP, 930
eQCWWAN_ERR_QMI_INVALID_TRANSITION, 928
eQCWWAN_ERR_QMI_INVALID_TX_ID, 927
eQCWWAN_ERR_QMI_MALFORMED_MSG, 927
eQCWWAN_ERR_QMI_MAX, 929
eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE, 928
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE, 928
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE, 928
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT, 928
eQCWWAN_ERR_QMI_MISSING_ARG, 927
eQCWWAN_ERR_QMI_MSG_BLOCKED, 929
eQCWWAN_ERR_QMI_NETWORK_ABORTED, 929
eQCWWAN_ERR_QMI_NETWORK_NOT_READY, 928
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE, 928
eQCWWAN_ERR_QMI_NO_EFFECT, 927
eQCWWAN_ERR_QMI_NO_ENTRY, 928
eQCWWAN_ERR_QMI_NO_FREE_PROFILE, 927
eQCWWAN_ERR_QMI_NO_MEMORY, 927
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND, 927
eQCWWAN_ERR_QMI_NO_RADIO, 929
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION, 929
eQCWWAN_ERR_QMI_NO_THRESHOLDS, 927
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE, 928
eQCWWAN_ERR_QMI_NOT_PROVISIONED, 927
eQCWWAN_ERR_QMI_NOT_SUPPORTED, 929
eQCWWAN_ERR_QMI_OFFSET, 927
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED, 927
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED, 927
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE, 929
eQCWWAN_ERR_QMI_OUT_OF_CALL, 927
eQCWWAN_ERR_QMI_PIN_BLOCKED, 928
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED, 928
eQCWWAN_ERR_QMI_POLICY_MISMATCH, 929
eQCWWAN_ERR_QMI_REQ, 926
eQCWWAN_ERR_QMI_REQ_SCH, 926
eQCWWAN_ERR_QMI_REQ_TO, 926
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED, 928
eQCWWAN_ERR_QMI_RSP, 926
eQCWWAN_ERR_QMI_RSP_TO, 926
eQCWWAN_ERR_QMI_SEGMENT_ORDER, 929
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG, 929
eQCWWAN_ERR_QMI_SESSION_INACTIVE, 928
eQCWWAN_ERR_QMI_SESSION_INVALID, 928
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP, 928
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND, 929
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED, 928
eQCWWAN_ERR_QMI_SMSC_ADDR, 929
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE, 929
eQCWWAN_ERR_QMI_TPDU_TYPE, 929
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION, 927
eQCWWAN_ERR_QMI_UNKNOWN, 928
eQCWWAN_ERR_QMI_WIDTH, 930
eQCWWAN_ERR_RESET, 927
eQCWWAN_ERR_SWICM_AM_VERS_ERROR, 929
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS, 929
eQCWWAN_ERR_SWICM_END, 930
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS, 929
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID, 929
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID, 929

eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID, 929
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED, 929
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED, 929
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED, 929
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS, 929
eQCWWAN_ERR_SWICM_SOCKET_IN_USE, 929
eQCWWAN_ERR_SWICM_START, 929
eQCWWAN_ERR_SWICM_TIMEOUT, 929
eQCWWAN_ERR_SWICM_V4DWN_V6DWN, 929
eQCWWAN_ERR_SWICM_V4DWN_V6UP, 929
eQCWWAN_ERR_SWICM_V4UP_V6DWN, 929
eQCWWAN_ERR_SWICM_V4UP_V6UP, 929
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED, 930
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND, 930
eQCWWAN_ERR_SWIDCS_END, 930
eQCWWAN_ERR_SWIDCS_FILEIO_ERR, 930
eQCWWAN_ERR_SWIDCS_IOCTL_ERR, 930
eQCWWAN_ERR_SWIDCS_START, 930
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE, 930
eQCWWAN_ERR_SWIIM_END, 930
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND, 930
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED, 930
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE, 930
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE, 930
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX, 930
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH, 930
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE, 930
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL, 930
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS, 930
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT, 930
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE, 930
eQCWWAN_ERR_SWIIM_INVALID_PATH, 930
eQCWWAN_ERR_SWIIM_OPENING_DIR, 930
eQCWWAN_ERR_SWIIM_OPENING_FILE, 930
eQCWWAN_ERR_SWIIM_START, 930
eQCWWAN_ERR_SWISM_END, 930
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND, 930
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED, 930
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG, 930
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED, 930
eQCWWAN_ERR_SWISMS_START, 930
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE, 931
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR, 931
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED, 931
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES, 931
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY, 931
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET, 931
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET, 931
eWDS_ERR_PROFILE_REG_END, 931
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID, 931
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED, 931
eWDS_ERR_PROFILE_REG_RESULT_FAIL, 931
eWDS_ERR_PROFILE_REG_RESULT_LIST_END, 931
qmerrno.h, 924
eQCWWANError, 926
qm_wds_ds_profile_extended_err_codes, 930
qmiSmsMessageList, 385
messageIndex, 385
messageTag, 385
qmiWDSDataBearerTechnology, 385
currentNetwork, 386
ratMask, 386
soMask, 386
qos.h, 931
pack_qos_SLQSQosGetNetworkStatus, 933

pack_qos_SLQSQosSwiReadApnExtraParams, 933
pack_qos_SLQSQosSwiReadDataStats, 934
pack_qos_SLQSSetQosEventCallback, 935
unpack_qos_SLQSQosGetNetworkStatus, 935
unpack_qos_SLQSQosSwiReadApnExtraParams, 936
unpack_qos_SLQSQosSwiReadDataStats, 936
unpack_qos_SLQSSetQosEventCallback, 937
unpack_qos_SLQSSetQosEventCallback_ind, 937
unpack_qos_SLQSSetQosNWStatusCallback_ind, 937
unpack_qos_SLQSSetQosPriEventCallback_ind, 938
unpack_qos_SLQSSetQosStatusCallback_ind, 939
qosDeliveryOrder
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
qosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 577
QosFlowInfo
unpack_qos_SLQSSetQosEventCallback_ind_t, 578

RAN
unpack_nas_GetServingNetwork_t, 515
RAT
nas_QmiNas3GppNetworkRAT, 189
RATMask
currNetworkInfo, 56
wds_currNetworkInfo, 714
RFBandInfoElements, 386
activeBandClass, 387
activeChannel, 387
radioInterface, 387
unpack_nas_GetRFInfo_t, 513
RFTlv
unpack_nas_SetEventReportInd_t, 519
RMAutoConnect
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
RPTlv
NASQmiCbkNasSystemSelPrefInd, 235
RRTlv
unpack_nas_SetEventReportInd_t, 519
RSRPThresListLen
nas_RSRPThresh, 193
RSRQThresListLen
nas_RSRQThresh, 194
RSSIThresListLen
nas_RSSIThresh, 194
RX_EC_IO
nas_NetworkStat1x, 172
RX_PWR
nas_NetworkStat1x, 172
nas_NetworkStatEVDO, 174
RXChan
nas_LTEInfo, 153
rXDroppedCount
unpack_wds_GetPacketStatus_t, 656
rXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 656
rXOkBytesCount
unpack_wds_GetPacketStatus_t, 656
rXPacketErrors
unpack_wds_GetPacketStatus_t, 656
rXPacketOverflows
unpack_wds_GetPacketStatus_t, 656
rXPacketSuccesses
unpack_wds_GetPacketStatus_t, 656
radio
unpack_nas_GetSignalStrengths_t, 516
radio_if
pack_nas_SLQSNasGetTxRxInfo_t, 284
radioIf
nas_ecioListElement, 133
nas_errorRateListElement, 134
nas_rsrqInformation, 193
nas_rxSignalStrengthListElement, 197
RadioIfaces
unpack_dms_GetDeviceCap_t, 445
unpack_dms_GetDeviceCapabilities_t, 446
unpack_nas_GetServingNetwork_t, 515
RadioIfacesSize
unpack_dms_GetDeviceCap_t, 445
unpack_nas_GetServingNetwork_t, 515
radioIfacesSize
unpack_dms_GetDeviceCapabilities_t, 446
radioInterface
nas_RFInfoTlv, 191
nas_roamIndList, 192
nas_servSystem, 200
nas_SignalStrengthTlv, 201
nas_timeInfo, 212
RFBandInfoElements, 387
radioInterfaceList
NASServingSystemInfo, 237
radioInterfaceNo
NASServingSystemInfo, 237
radioInterfaceSize
nas_RFInfoTlv, 191
range
unpack_qos_Port_t, 572
rat
nas_CSGID, 125
nas_MNRInfo, 169
ratMask
qmiWDSDataBearerTechnology, 386
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
rawHorConfidence
pack_loc_SLQSLOCInjectPosition_t, 267
rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 267
rawLen
uim_fileAttributes, 425

rawValue
uim_fileAttributes, 425
readTransparent
pack_uim_ReadTransparent_t, 330
Reason
pack_voice_SLQSVoiceGetCallForwardingStatus-_t, 347
pack_voice_SLQSVoiceSetCallBarringPassword_t, 354
reason
pack_voice_SLQSVoiceGetCallBarring_t, 347
pack_voice_SLQSVoiceSetSUPSService_t, 358
unpack_qos_SLQSSetQosStatusCallback_ind_t, 580
voice_ccSUPSType, 694
receiptAction
sms_routeEntry, 401
reconfigReqd
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 669
recordCount
uim_fileAttributes, 425
recordSize
uim_fileAttributes, 425
RedirPartyNum
voice_arrRedirPartyNum, 684
refData
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 475
refID
pack_cat_CATSendTerminalResponse_t, 240
refString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 475
ReferenceID
cat_AlPhaIdentifierTlv, 48
cat_EventIDDataTlv, 50
refpn
nas_CDMAInfo, 117
refreshComplete
pack_uim_SLQSUIMRefreshComplete_t, 336
RefreshMode
cat_RefreshTlv, 51
RefreshStage
cat_RefreshTlv, 51
regAction
pack_nas_SLQSInitiateNetworkRegistration_t, 279
RegForeignNID
unpack_nas_GetCDMANetworkParameters_t, 508
RegForeignSID
unpack_nas_GetCDMANetworkParameters_t, 508
RegHomeSID
unpack_nas_GetCDMANetworkParameters_t, 508
regPrd
nas_AddCDMASysInfo, 112
regRefresh
pack_uim_SLQSUIMRefreshRegister_t, 337
regRejectInfoValid
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_WCDMASysInfo, 224
regState
nas_servSystem, 200
registerFlag
uim_registerRefresh, 431
RegistrationState
unpack_nas_GetServingNetwork_t, 515
registrationState
NASServingSystemInfo, 237
rejCause
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_WCDMASysInfo, 224
rejectCause
nas_RejectReasonTlv, 190
rejectSrvDomain
nas_GSMSysInfo, 142
nas_LTESysInfo, 166
nas_WCDMASysInfo, 224
reliabilityClass
LibPackGPRSRequestedQoS, 68
wds_GPRSQoS, 721
remPartyNumber
voice_remotePartyNum, 710
RemotePartyName
voice_getAllCallRmtPtyName, 703
RemotePartyNum
voice_getAllCallRmtPtyNum, 704
resBerRatio
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
ResCode
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 476
resetInfoInd
pack_dms_SLQSDmsSwiIndicationRegister_t, 248
resultcode
_unpack_swiavms_SLQSAVMSEventReportInd_t, 33
_unpack_swiavms_SLQSAVMSSessionGetInfo_t, 34
unpack_swiavms_SLQSAVMSGetSettings_t, 601
unpack_swiavms_SLQSAVMSSendSelection_t, 601
unpack_swiavms_SLQSAvmsSetEventReport_t, 602
unpack_swiavms_SLQSAVMSSetSettings_t, 602
unpack_swiavms_SLQSAVMSStartSession_t, 602
unpack_swiavms_SLQSAVMSStopSession_t, 603
RetryCount
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
revTunneling
unpack_wds_GetMobileIPProfile_t, 653
ReverseMac
nas_protocolSubtypeElement, 187
RfSarState
pack_sar_SLQSSetRfSarState_t, 311
RmTrasnferStaticsReq

pack_wds_RMSetTransferStatistics_t, 363
rmTrasnferStaticsReq, 387
bResetStatistics, 388
ulMask, 388
rms.h, 939
pack_rms_GetSMSWake, 940
pack_rms_SetSMSWake, 940
unpack_rms_GetSMSWake, 940
unpack_rms_SetSMSWake, 941
RmtPtyNum
voice_arrRemotePartyNum, 685
roamIndicator
nas_roamIndList, 192
RoamIndicatorVal
unpack_nas_SLQSGetServingSystem_t, 529
roamOrigVoiceSO
voice_prefVoiceSO, 709
RoamPref
NASRoamPreferenceTlv, 235
roamStatus
nas_sysInfoCommon, 207
roamStatusValid
nas_sysInfoCommon, 207
roamTimerValue
voice_roamTimer, 711
Roaming
nas_QmiNas3GppNetworkInfo, 189
unpack_nas_GetCDMANetworkParameters_t, 508
unpack_nas_GetServingNetwork_t, 515
roaming
unpack_nas_SetRoamingIndicatorCallback_ind_t, 521
RoamingIndicatorList
unpack_nas_SLQSGetServingSystem_t, 529
routeList
sms_setRoutesReq, 405
routeStorage
sms_routeEntry, 401
rptPosData
pack_pds_SetEventReportCallback_t, 299
rptRate
nas_LTESigRptCfg, 161
nas_LTESigRptConfig, 162
rscp
nas_rxInfo, 195
nas_UMTSInfo, 215
tdscdmaSigInfoExt, 411
rsrp
lteSSInfo, 110
nas_cellParams, 123
nas_rxInfo, 195
nas_RxSigInfo, 196
nas_umtsLTENbrCell, 216
rsrplevel
nas_lteRsrpinformation, 159
rsrq
lteSSInfo, 110
nas_cellParams, 123
nas_rsrqInformation, 193
nas_SccRxInfo, 198
nas_umtsLTENbrCell, 217
rsrqDelta
nas_SLQSSignalStrengthsIndReq, 203
rsrqInfo
nas_SLQSSignalStrengthsInformation, 204
unpack_nas_SLQSGetSignalStrength_t, 531
rssi
cdmaSSInfo, 52
hdrSSInfo, 66
lteSSInfo, 110
nas_cellParams, 123
nas_gsmCellInfo, 138
tdscdmaSigInfoExt, 411
unpack_nas_GetSignalStrengths_t, 516
rx_bytes
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
rx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
rxChainIndex
nas_RxSigInfo, 196
RxDropConutTlv
unpack_RMTransferStatistics_ind_t, 591
rxLev
nas_GERANInfo, 136
rxOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 664
RxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 591
RxOkConutTlv
unpack_RMTransferStatistics_ind_t, 591
rxPower
nas_rxInfo, 196
nas_RxSigInfo, 197
RxQFilter
unpack_qos_QosFlowInfo_t, 574
RxQFlowGranted
unpack_qos_QosFlowInfo_t, 574
rxSignalStrength
nas_rxSignalStrengthListElement, 197
rxSignalStrengthDelta
nas_SLQSSignalStrengthsIndReq, 203
rxSignalStrengthInfo
nas_SLQSSignalStrengthsInformation, 204
rxSignalStrengthList
unpack_nas_SLQSGetSignalStrength_t, 531
rxSignalStrengthListLen
unpack_nas_SLQSGetSignalStrength_t, 531

SCI
unpack_nas_GetCDMANetworkParameters_t, 508
SCM
unpack_nas_GetCDMANetworkParameters_t, 508
SDPTlv
NASQmiCbkNasSystemSelPrefInd, 235
SDU_HDR_LEN

common.h, 736
SHORT
SwiDataTypes.h, 984
SI
voice_callFWExtInfo, 691
sIntraSearch
nas_LTEInfoIntrafreq, 156
SLQSSSTlv
unpack_nas_SetEventReportInd_t, 519
SMS_CONFIG_LEN
sms.h, 946
SMS_MAX_SMS_ROUTES
sms.h, 946
SMS_NUM_OF_SET
sms.h, 946
SMSC_TYPE_LEN
sms.h, 946
sMSCAddress, 406
data, 406
length, 406
sMSCAddressInfo
sms.h, 947
sMSCAddressTlv, 406
SMSCInfo, 406
TlvPresent, 406
SMSCInfo
sMSCAddressTlv, 406
SMSCTlv
unpack_sms_SetNewSMSCallback_ind_t, 594
sMSEtwsMessage, 407
data, 407
length, 407
notificationType, 407
sMSEtwsMessageInfo
sms.h, 947
sMSEtwsMessageTlv, 407
EtwsMessageInfo, 407
TlvPresent, 407
sMSEtwsPlmn, 408
mobileCountryCode, 408
mobileNetworkCode, 408
sMSEtwsPlmnInfo
sms.h, 947
sMSMTMessage, 408
messageIndex, 409
storageType, 409
sMSMTMessageInfo
sms.h, 947
sMSMessageMode, 408
messageMode, 408
sMSMessageModeInfo
sms.h, 947
sMSOnIMS, 409
smsOnIMS, 409
sMSOnIMSInfo
sms.h, 947
sMSOnIMSTlv, 409
IMSInfo, 409
TlvPresent, 409
SMSSupport
pack_dms_SetCustFeature_t, 245
unpack_dms_GetCustFeature_t, 443
sMSTransferRouteMTMessage, 410
ackIndicator, 410
data, 410
format, 410
length, 410
transactionID, 410
sMSTransferRouteMTMessageInfo
sms.h, 948
SNR
nas_NetworkStatEVDO, 174
sNonIntraSearch
nas_LTEInfoIntrafreq, 156
SO
nas_NetworkStat1x, 172
SOMask
currNetworkInfo, 56
wds_currNetworkInfo, 714
SPC_SIZE
dms.h, 747
sPhyCaAggPcellInfo
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 519
sPhyCaAggScellDlBw
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 519
sPhyCaAggScellIndType
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 519
sPhyCaAggScellIndex
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 519
sPhyCaAggScellInfo
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 519
SSInfo
unpack_nas_SetServingSystemCallback_ind_t, 521
sSLQSSignalStrengthsInfo
nas_SLQSSignalStrengthsTlv, 204
SSTlv
unpack_nas_SetEventReportInd_t, 519
SUPSType
pack_voice_SLQSVoiceManageCalls_t, 352
SWI_API
SwiDataTypes.h, 983
SWIWWANCMAPI.h, 995
sar.h, 941
pack_sar_SLQSGetRfSarState, 941
pack_sar_SLQSSetRfSarState, 942
unpack_sar_SLQSGetRfSarState, 942
unpack_sar_SLQSSetRfSarState, 942
sbas_almanac_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 569
sbas_ephemeris_sv_msk

unpack_pds_SLQSGetGPSStateInfo_t, 569
sbas_health_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 569
sbas_visible_sv_msk
unpack_pds_SLQSGetGPSStateInfo_t, 569
scell_idx
nas_PhyCaAggScellIndex, 179
NASPhyCaAggScellIndex, 231
scell_state
nas_PhyCaAggScellIndType, 179
nas_PhyCaAggScellInfo, 182
NASPhyCaAggScellIndType, 232
NASPhyCaAggScellInfo, 233
screeningInd
voice_connectNumInfo, 700
sduErrorRatio
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
seDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 675
seDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 675
sePCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 675
sePCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 675
secActivate
uim_fileAttributes, 425
secActivateMask
uim_fileAttributes, 425
secChA
nas_CDMAChannel, 115
secChB
nas_CDMAChannel, 115
secDeactivate
uim_fileAttributes, 425
secDeactivateMask
uim_fileAttributes, 425
secIncrease
uim_fileAttributes, 425
secIncreaseMask
uim_fileAttributes, 425
SecProt
nas_protocolSubtypeElement, 187
secRead
uim_fileAttributes, 425
secReadMask
uim_fileAttributes, 425
secWrite
uim_fileAttributes, 425
secWriteMask
uim_fileAttributes, 425
secdns
unpack_wds_GetDefaultProfile_t, 649
secdnsv6
unpack_wds_GetDefaultProfile_t, 649
second
nas_timeInfo, 212
nas_UniversalTime, 218
secondaryDNS
pack_wds_SetDefaultProfile_t, 365
SecondaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 667
SecondaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 667
secondaryHA
unpack_wds_GetMobileIPProfile_t, 653
SectorIDLen
nas_NetworkStatEVDO, 174
selNetwork
nas_servSystem, 200
selected
sms_BroadcastConfig, 393
sms_CDMABroadcastConfig, 394
selectedNetwork
NASServingSystemInfo, 237
selection
pack_swiavms_SLQSAVMSSendSelection_t, 321
pack_swioma_SLQSOMADMSendSelection_t, 327
sensorData_t, 388
flags, 389
sensorDataLen, 389
timeOfFirstSample, 389
timeOffset, 389
xAxis, 389
yAxis, 389
zAxis, 389
sensorDataLen
sensorData_t, 389
ServerAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 667
serverAddrTypeMask
pack_loc_SLQSLOCGetServer_t, 262
serverStatus
unpack_loc_GetServer_Ind_t, 495
serverType
pack_loc_SLQSLOCGetServer_t, 262
unpack_loc_GetServer_Ind_t, 495
serviceCategory
sms_CDMABroadcastConfig, 394
serviceDomain
nas_RejectReasonTlv, 190
servingCellId
nas_LTEInfoIntrafreq, 156
ServingSystem
unpack_nas_SLQSGetServingSystem_t, 529
sessionControl
pack_pds_StartPDSTrackingSessionExt_t, 308
sessionEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 669

SessionId
pack_loc_Start_t, 272
pack_loc_Stop_t, 272
sessionId
unpack_loc_PositionRpt_Ind_t, 502
sessionInfo
pack_uim_ChangePin_t, 329
pack_uim_ReadTransparent_t, 331
pack_uim_SetPinProtection_t, 332
pack_uim_SLQSUIMAuthenticate_t, 332
pack_uim_SLQSUIMGetFileAttributes_t, 334
pack_uim_SLQSUIMRefreshComplete_t, 336
pack_uim_SLQSUIMRefreshGetLastEvent_t, 336
pack_uim_SLQSUIMRefreshOK_t, 337
pack_uim_SLQSUIMRefreshRegister_t, 337
pack_uim_UnblockPin_t, 339
pack_uim_VerifyPin_t, 340
SessionInfoConfig
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 605
SessionInfoFota
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 605
SessionInfoNotification
unpack_swioma_SLQSOMADMAlertCallback_ind-_t, 605
sessionOperation
pack_pds_StartPDSTrackingSessionExt_t, 308
sessionResponse
unpack_swiavms_SLQSAVMSStartSession_t, 603
sessionServerOption
pack_pds_StartPDSTrackingSessionExt_t, 308
SessionState
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
SessionStatus
unpack_pds_SetEventReport_Ind_t, 564
sessionStatus
unpack_loc_PositionRpt_Ind_t, 502
unpack_omaDmNotificationsTlv_t, 558
SessionType
pack_swioma_SLQSOMADMGetSessionInfo_t, 326
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
sessionType
pack_pds_StartPDSTrackingSessionExt_t, 308
pack_swiavms_SLQSAVMSStartSession_t, 323
pack_swiavms_SLQSAVMSStopSession_t, 324
pack_swioma_SLQSOMADMCancelSession_t, 326
pack_swioma_SLQSOMADMStartSession_t, 328
uim_refreshevent, 430
uim_sessionInformation, 432
uim_UIMSessionInformation, 436
unpack_omaDmFotaTlv_t, 557
set_fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 325
set_fix_type
pack_swiloc_SwiLocSetAutoStart_t, 325
set_function
pack_swiloc_SwiLocSetAutoStart_t, 325
set_max_dist
pack_swiloc_SwiLocSetAutoStart_t, 325
set_max_time
pack_swiloc_SwiLocSetAutoStart_t, 325
SetupEventList
cat_EventListTlv, 51
Severity
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
severity
unpack_omaDmFotaTlv_t, 557
shortName
nas_PLMNNetworkNameData, 185
unpack_nas_SLQSGetPLMNName_t, 525
shortNameCI
unpack_nas_SLQSGetPLMNName_t, 525
shortNameEn
unpack_nas_SLQSGetPLMNName_t, 525
shortNameLen
nas_PLMNNetworkNameData, 185
unpack_nas_SLQSGetPLMNName_t, 525
shortNameSB
unpack_nas_SLQSGetPLMNName_t, 525
shortNameSpareBits
nas_PLMNNetworkNameData, 185
sid
nas_CDMAInfo, 117
nas_sidNid, 201
unpack_nas_GetHomeNetwork_t, 511
SidNid
nas_homeSIDNID, 147
SigInd
LibPackUMTSReqQoSSigInd, 96
sigInfo
nas_SccRxInfo, 198
signalStrength
nas_SignalStrengthTlv, 201
signalStrengthReqMask
unpack_nas_SLQSGetSignalStrength_t, 531
SimCapability
unpack_dms_GetDeviceCap_t, 445
simCapability
unpack_dms_GetDeviceCapabilities_t, 446
sinr
hdrSSInfo, 66
nas_SLQSSignalStrengthsInformation, 204
tdscdmaSigInfoExt, 411
unpack_nas_SLQSGetSignalStrength_t, 531
sinrDelta
nas_SLQSSignalStrengthsIndReq, 203
sinrThresholdList
nas_SLQSSignalStrengthsIndReq, 203
sinrThresholdListLen
nas_SLQSSignalStrengthsIndReq, 203

sku_str
unpack_dms_GetFirmwareInfo_t, 450
slot
pack_uim_SLQSUIMPowerDown_t, 335
pack_uim_SLQSUIMPowerUp_t, 335
slot_t, 390
bICCID, 390
bICCIDLength, 390
bLogicalSlot, 390
uPhyCardStatus, 390
uPhySlotStatus, 390
slotInf, 391
AppStatus, 392
cardState, 392
errorState, 392
numApp, 392
upinRetries, 392
upinState, 392
upukRetries, 392
SlotInfo
uim_cardStatus, 419
slots_t, 392
uimSlotStatus, 392
slotsstatusChange
unpack_uim_SetUimSlotStatusChangeCallback_-ind_t, 613
SlqsProfile3GPP
unpackWdsProfileParam, 676
wds_profileInfo, 725
SlqsProfile3GPP2
unpackWdsProfileParam, 676
wds_profileInfo, 725
sms.h
LITEQMI_QMI_CBK_PARAM_NOCHANGE, 948
LITEQMI_QMI_CBK_PARAM_RESET, 948
LITEQMI_QMI_CBK_PARAM_SET, 948
sms.h, 943
eqmiCbkSetStatus, 948
MAX_MSE_TWS_MSG, 946
MAX_SMS_LIST_SIZE, 946
pack_sms_GetSMSCAddress, 948
pack_sms_SLQSDeleteSMS, 950
pack_sms_SLQSGetIndicationRegister, 950
pack_sms_SLQSGetMessageWaiting, 951
pack_sms_SLQSGetSMS, 951
pack_sms_SLQSGetSMSList, 952
pack_sms_SLQSGetSmsBroadcastConfig, 951
pack_sms_SLQSGetTransLayerInfo, 952
pack_sms_SLQSGetTransNWRegInfo, 953
pack_sms_SLQSModifySMSStatus, 953
pack_sms_SLQSSendAsyncSMS, 953
pack_sms_SLQSSetIndicationRegister, 954
pack_sms_SLQSSetSmsBroadcastActivation, 954
pack_sms_SLQSSetSmsBroadcastConfig, 954
pack_sms_SLQSSetSmsStorage, 955
pack_sms_SLQSSmsGetMaxStorageSize, 955
pack_sms_SLQSSmsGetMessageProtocol, 955
pack_sms_SLQSSmsSetRoutes, 956
pack_sms_SLQSSwiGetSMSStorage, 956
pack_sms_SaveSMS, 948
pack_sms_SendSMS, 949
pack_sms_SetNewSMSCallback, 949
pack_sms_SetSMSCAddress, 950
SMS_CONFIG_LEN, 946
SMS_MAX_SMS_ROUTES, 946
SMS_NUM_OF_SET, 946
SMSC_TYPE_LEN, 946
sMSCAddressInfo, 947
sMSEtwsMessageInfo, 947
sMSEtwsPlmnInfo, 947
sMSMTMessageInfo, 947
sMSMessageModeInfo, 947
sMSOnIMSInfo, 947
sMSTransferRouteMTMessageInfo, 948
unpack_sms_GetSMSCAddress, 957
unpack_sms_SLQSDeleteSMS, 959
unpack_sms_SLQSGetIndicationRegister, 959
unpack_sms_SLQSGetMessageWaiting, 959
unpack_sms_SLQSGetSMS, 960
unpack_sms_SLQSGetSMSList, 960
unpack_sms_SLQSGetSmsBroadcastConfig, 960
unpack_sms_SLQSGetTransLayerInfo, 961
unpack_sms_SLQSGetTransNWRegInfo, 961
unpack_sms_SLQSModifySMSStatus, 961
unpack_sms_SLQSSendAsyncSMS, 962
unpack_sms_SLQSSetIndicationRegister, 962
unpack_sms_SLQSSetSmsBroadcastActivation, 962
unpack_sms_SLQSSetSmsBroadcastConfig, 963
unpack_sms_SLQSSetSmsStorage, 963
unpack_sms_SLQSSmsGetMaxStorageSize, 963
unpack_sms_SLQSSmsGetMessageProtocol, 964
unpack_sms_SLQSSmsSetRoutes, 964
unpack_sms_SLQSSwiGetSMSStorage, 964
unpack_sms_SLQSWmsMemoryFullCallBack_ind, 965
unpack_sms_SaveSMS, 957
unpack_sms_SendSMS, 957
unpack_sms_SetNewSMSCallback, 958
unpack_sms_SetNewSMSCallback_ind, 958
unpack_sms_SetSMSCAddress, 958
sms_BroadcastConfig, 392
fromServiceId, 393
selected, 393
toServiceId, 393
sms_CDMABroadcastConfig, 393
language, 394
selected, 394
serviceCategory, 394
sms_getIndicationReg, 394
pRegCallStatInfoEvt, 395
pRegTransLayerInfoEvt, 395
pRegTransNWRegInfoEvt, 395
sms_getMsgWaitingInfo, 395
msgWaitInfo, 395
numInstances, 395

sms_getTransLayerInfo, 395
pRegInd, 396
pTransLayerInfo, 396
sms_getTransNWRegInfo, 396
pRegStatus, 396
sms_maxStorageSizeReq, 397
pMessageMode, 397
storageType, 397
sms_maxStorageSizeResp, 397
freeSlots, 398
maxStorageSize, 398
sms_messageWaitingInfoContent, 398
activeInd, 398
msgCount, 398
msgType, 398
sms_msgProtocolResp, 399
msgProtocol, 399
sms_qaQmi3GPP2BroadcastCfgInfo, 399
activated_ind, 400
CDMABroadcastConfig, 400
num_instances, 400
sms_qaQmi3GPPBroadcastCfgInfo, 400
activated_ind, 400
broadcastConfig, 400
num_instances, 400
sms_routeEntry, 400
messageClass, 401
messageType, 401
receiptAction, 401
routeStorage, 401
sms_sendAsyncsmsParams, 401
messageFormat, 403
messageSize, 403
pFollowOnDC, 403
pForceOnDC, 403
pLinktimer, 403
pMessage, 403
pRetryMessage, 403
pRetryMessageId, 403
pServiceOption, 403
pSmsOnIms, 403
pUserData, 403
sms_setIndicationReg, 404
pRegCallStatInfoEvt, 404
pRegTransLayerInfoEvt, 404
pRegTransNWRegInfoEvt, 404
sms_setRoutesReq, 404
numOfRoutes, 405
pTransferStatusReport, 405
routeList, 405
sms_transLayerInfo, 405
TransCap, 405
TransType, 406
smsOnIMS
sMSOnIMS, 409
smsStorage
pack_sms_SLQSSetSmsStorage_t, 319
snr
loc_satelliteInfo, 106
lteSSInfo, 110
nas_SccRxInfo, 198
snrlevel
nas_lteSnrinformation, 162
soMask
qmiWDSDataBearerTechnology, 386
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
Source
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
source
altSrcInfo_t, 43
unpack_dms_GetNetworkTime_t, 455
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 466
unpack_dms_SLQSDmsSwiGetResetInfo_t, 467
sourceIPMask
LibPackTFTIDParams, 93
SourceLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
spc
pack_dms_ResetToFactoryDefaults_t, 242
pack_nas_SetACCOLC_t, 273
pack_wds_SetMobileIPProfile_t, 369
spn
nas_serviceProviderName, 199
unpack_nas_SLQSGetPLMNName_t, 525
spnEncoding
unpack_nas_SLQSGetPLMNName_t, 525
spnLength
nas_serviceProviderName, 199
unpack_nas_SLQSGetPLMNName_t, 525
srcPortRangeEnd
LibPackTFTIDParams, 93
srcPortRangeStart
LibPackTFTIDParams, 93
srvCapability
nas_detailSvcInfo, 130
nas_sysInfoCommon, 207
srvCapabilityValid
nas_sysInfoCommon, 207
srvDomain
nas_sysInfoCommon, 207
SrvDomainPref
NASServDomainPrefTlv, 236
srvDomainValid
nas_sysInfoCommon, 207
srvOption
voice_arrSvcOption, 686
srvStatus
nas_detailSvcInfo, 130
nas_GSMSrvStatusInfo, 139
nas_SrvStatusInfo, 205
srxlev
nas_cellParams, 123

nas_gsmCellInfo, 138
nas_umtsLTENbrCell, 217
nas_wcdmaCellInfo, 219
stage
uim_refreshevent, 430
State
nas_NetworkStat1x, 172
nas_NetworkStatEVDO, 174
state
pack_loc_SLQSLOCSetCradleMountConfig_t, 270
unpack_dms_GetActivationState_t, 439
unpack_dms_SLQSUIMGetState_t, 479
unpack_omaDmConfigTlv_t, 555
unpack_omaDmFotaTlv_t, 557
unpack_qos_QosFlowInfoState_t, 575
statmask
pack_wds_GetPacketStatus_t, 362
StatsMask
transferStatInd, 413
statsMask
wds_TrStatInd, 726
StatsPeriod
transferStatInd, 413
statsPeriod
wds_TrStatInd, 726
Status
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
status
nas_lteEARFCN, 150
nas_ltePCI, 158
nas_wcdmaUARFCN, 224
pack_sms_SetNewSMSCallback_t, 313
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_DeleteAssistData_Ind_t, 492
unpack_loc_SetExtPowerConfig_Ind_t, 503
unpack_loc_SetOperationMode_Ind_t, 504
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 578
unpack_qos_SLQSSetQosStatusCallback_ind_t, 580
storageIndex
FMSImageIdElement, 63
storageType
pack_sms_SaveSMS_t, 312
pack_sms_SLQSDeleteSMS_t, 314
pack_sms_SLQSGetSMS_t, 315
pack_sms_SLQSGetSMSList_t, 316
pack_sms_SLQSModifySMSStatus_t, 317
sms_maxStorageSizeReq, 397
sMSMTMessage, 409
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 600
Streaming Download Protocol (lite-fw), 26
String
unpack_dms_GetDeviceHardwareRev_t, 447
unpack_dms_GetDeviceMfr_t, 447
unpack_dms_GetFSN_t, 453
unpack_dms_UIMGetICCID_t, 481
stringSize
unpack_dms_GetDeviceHardwareRev_t, 447
unpack_dms_GetDeviceMfr_t, 448
unpack_dms_UIMGetICCID_t, 481
subAddr
voice_calledPartySubAdd, 688
subAddrLen
voice_calledPartySubAdd, 688
subAddrType
voice_calledPartySubAdd, 688
subnetMask
unpack_qos_IPv4Addr_t, 571
SubnetMaskV4
unpack_wds_SLQSGetRuntimeSettings_t, 667
subsType
pack_voice_SLQSVoiceBindSubscription_t, 343
SupUSBComps
unpack_dms_GetUSBComp_t, 460
SuppOA
voice_CUGInfo, 700
SuppPrefCUG
voice_CUGInfo, 700
svInfoMask
loc_satelliteInfo, 106
svListLen
loc_satelliteInfo, 106
svStatus
loc_satelliteInfo, 106
svc
pack_qmi_t, 309
SvcClass
voice_callFWExtInfo, 691
voice_callFWInfo, 692
SvcStatus
voice_callFWExtInfo, 691
voice_callFWInfo, 692
svcType
voice_ccSUPSType, 694
sw1
uim_cardResult, 418
sw2
uim_cardResult, 418
swVerString
pack_dms_SLQSSwiSetHostDevInfo_t, 250
unpack_dms_SLQSSwiGetHostDevInfo_t, 476
SwiDataTypes.h, 983
BOOL, 984
BYTE, 984
CHAR, 984
FLOAT, 984
INT32, 984
INT8, 984
LPCSTR, 984
SHORT, 984
SWI_API, 983
ULONG, 984
ULONGLONG, 984

UNUSEDPARAM, 983
USHORT, 984
WORD, 984
swiavms.h, 965
pack_swiavms_SLQSAVMSGetSettings, 974
pack_swiavms_SLQSAVMSSendSelection, 975
pack_swiavms_SLQSAVMSSessionGetInfo, 975
pack_swiavms_SLQSAVMSSetSettings, 976
pack_swiavms_SLQSAVMSSetSettingsNoAutoRebootField, 977
pack_swiavms_SLQSAVMSStartSession, 978
pack_swiavms_SLQSAVMSStopSession, 978
pack_swiavms_SLQSAvmsSetEventReport, 976
PackSwiAVMSSettingsAPNInfo, 968
PackSwiAVMSSettingsConnectionRetryTimers, 968
PackSwiAVMSSettingsPeriodsInfo, 968
PackSwiAvmsSetSettingsAPNInfo, 968
PackSwiAvmsSetSettingsConnectionRetryTimers, 968
PackSwiAvmsSetSettingsPeriodInfo, 968
unpack_swiavms_SLQSAVMSEventReportInd, 979
unpack_swiavms_SLQSAVMSEventReportInd_t, 968
unpack_swiavms_SLQSAVMSGetSettings, 979
unpack_swiavms_SLQSAVMSSendSelection, 980
unpack_swiavms_SLQSAVMSSessionGetInfo, 980
unpack_swiavms_SLQSAVMSSessionGetInfo_t, 969
unpack_swiavms_SLQSAVMSSetSettings, 981
unpack_swiavms_SLQSAVMSStartSession, 982
unpack_swiavms_SLQSAVMSStopSession, 982
unpack_swiavms_SLQSAvmsSetEventReport, 981
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 969
UnpackSwiAvmsEventReportConfig, 970
UnpackSwiAvmsEventReportConnectionRequest, 970
UnpackSwiAvmsEventReportDataSessionStatus, 971
UnpackSwiAvmsEventReportHTTPStatus, 971
UnpackSwiAvmsEventReportNotification, 972
UnpackSwiAvmsEventReportPackageID, 973
UnpackSwiAvmsEventReportRegStatus, 973
UnpackSwiAvmsEventReportSessionType, 973
UnpackSwiAvmsEventReportWAMSParamChange, 974
swiloc.h, 984
pack_swiloc_SwiLocGetAutoStart, 984
pack_swiloc_SwiLocSetAutoStart, 985
unpack_swiloc_SwiLocGetAutoStart, 985
unpack_swiloc_SwiLocSetAutoStart, 985
swioma.h, 986
pack_swioma_SLQSOMADMAlertCallback, 987
pack_swioma_SLQSOMADMCancelSession, 987
pack_swioma_SLQSOMADMGetSessionInfo, 988
pack_swioma_SLQSOMADMGetSettings, 989
pack_swioma_SLQSOMADMSendSelection, 989
pack_swioma_SLQSOMADMSetSettings, 990
pack_swioma_SLQSOMADMStartSession, 991
unpack_swioma_SLQSOMADMAlertCallback, 991
unpack_swioma_SLQSOMADMAlertCallback_ind, 992
unpack_swioma_SLQSOMADMCancelSession, 992
unpack_swioma_SLQSOMADMGetSessionInfo, 993
unpack_swioma_SLQSOMADMGetSettings, 993
unpack_swioma_SLQSOMADMSendSelection, 993
unpack_swioma_SLQSOMADMSetSettings, 994
unpack_swioma_SLQSOMADMStartSession, 994
switchOption
pack_voice_SLQSVoiceALSSetLineSwitching_t, 342
sysInfoCDMA
nas_CDMASysInfo, 121
sysInfoGSM
nas_GSMSysInfo, 142
sysInfoHDR
nas_HDRSysInfo, 147
sysInfoLTE
nas_LTESysInfo, 166
sysInfoWCDMA
nas_WCDMASysInfo, 224
system
loc_satelliteInfo, 106
loc_SV, 108
systemDiscontinuities
pack_pds_PDSInjectTimeReference_t, 297
SystemID
unpack_nas_SLQSGetServingSystem_t, 529
systemID
nas_CDMASysInfo, 121
systemMode
nas_CommInfo, 124
systemTime
pack_pds_PDSInjectTimeReference_t, 297
szAPN
_PackSwiAvmsSetSettingsAPNInfo, 30
szAlertMsg
_UnpackSwiAvmsEventReportConfig, 37
szCarrier_str
_litefw_FirmwareInfo_, 29
szCarrierPriversion_str
_litefw_FirmwareInfo_, 30
szDescription
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 35
szFwversion_str
_litefw_FirmwareInfo_, 30
szModelid_str
_litefw_FirmwareInfo_, 30
szName

_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 35
szPWD
_PackSwiAvmsSetSettingsAPNInfo, 30
szPackageid_str
_litefw_FirmwareInfo_, 30
szSku_str
_litefw_FirmwareInfo_, 30
szUname
_PackSwiAvmsSetSettingsAPNInfo, 30
szVersion
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36

t3396_apn
unpack_nas_SLQSNasTimerCallback_ind_t, 547
t3396_plmn_id
unpack_nas_SLQSNasTimerCallback_ind_t, 547
t3396_val
unpack_nas_SLQSNasTimerCallback_ind_t, 547
TCPDstPort
unpack_qos_swiQosFilter_t, 583
TCPSrcPort
unpack_qos_swiQosFilter_t, 584
TDSCDMAECIOThreshListLen
nas_TDSCDMAECIOThresh, 208
TDSCDMARSCPThreshListLen
nas_TDSCDMARSCPThresh, 208
TDSCDMARSSIThreshListLen
nas_TDSCDMARSSIThresh, 209
TDSCDMASINRThreshListLen
nas_TDSCDMASINRThresh, 210
TRMessageTlv
unpack_sms_SetNewSMSCallback_ind_t, 595
TX_PWR
nas_NetworkStat1x, 172
TXChan
nas_LTEInfo, 153
tXDroppedCount
unpack_wds_GetPacketStatus_t, 656
tXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 656
tXOkBytesCount
unpack_wds_GetPacketStatus_t, 656
tXPacketErrors
unpack_wds_GetPacketStatus_t, 656
tXPacketOverflows
unpack_wds_GetPacketStatus_t, 656
tXPacketSuccesses
unpack_wds_GetPacketStatus_t, 656
Tables, 27
tac
nas_LTEInfoIntrafreq, 156
nas_LTESysInfo, 166
tacValid
nas_LTESysInfo, 166
TdsBandCapability
unpack_dms_SLQSGetBandCapability_t, 471
tdscdmaSigInfoExt, 410
ecio, 411
rscp, 411
rssi, 411
sinr, 411
techName
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 669
Technology
nas_DeviceConfigDetail, 131
unpack_wds_SLQSGetRuntimeSettings_t, 667
TechnologyPref
pack_nas_SetNetworkPreference_t, 275
tempData_t, 411
temperature, 412
temperatureDataLen, 412
timeOfFirstSample, 412
timeOffset, 412
timeSource, 412
temperature
nas_CommInfo, 124
tempData_t, 412
temperatureDataLen
tempData_t, 412
threshGsmHigh
nas_lteGsmCellInfo, 151
threshGsmLow
nas_lteGsmCellInfo, 151
threshServingLow
nas_LTEInfoIntrafreq, 156
threshXHigh
nas_infoInterFreq, 149
threshXLow
nas_infoInterFreq, 149
threshXhigh
nas_lteWcdmaCellInfo, 167
threshXlow
nas_lteWcdmaCellInfo, 167
Time
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
time
NASTimeInfoTlv, 238
Time_uncert_ms
unpack_pds_SLQSGetGPSStateInfo_t, 569
timeBase
pack_pds_SLQSPDSInjectAbsoluteTimeReference-_t, 303
TimeLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
timeMsec
pack_loc_SLQSLOCInjectUTCTime_t, 269
pack_pds_SLQSPDSInjectAbsoluteTimeReference-_t, 303
timeOfFirstSample
sensorData_t, 389
tempData_t, 412
timeOffset

sensorData_t, 389
tempData_t, 412
timeSource
tempData_t, 412
TimeStmp_gps_week
unpack_pds_SLQSGetGPSStateInfo_t, 569
TimeStmp_tow_ms
unpack_pds_SLQSGetGPSStateInfo_t, 569
timeTlv
NASQmiCbkNasSwiOTAMessageInd, 234
timeUncMsec
pack_loc_SLQSLOCInjectUTCTime_t, 269
pack_pds_SLQSPDSInjectAbsoluteTimeReference_t, 303
timeZone
nas_timeInfo, 212
timeout
pack_pds_SetPDSDefaults_t, 300
pack_qmi_t, 309
Timers
_PackSwiAVMSSettingsConnectionRetryTimers, 32
_PackSwiAvmsSetSettingsConnectionRetryTimers, 31
timestamp
unpack_dms_GetNetworkTime_t, 455
timestampAge
pack_loc_SLQSLOCInjectPosition_t, 267
timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 267
timingAdvance
nas_GERANInfo, 136
TlvPresent
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 36
_UnpackSwiAvmsEventReportConfig, 37
_UnpackSwiAvmsEventReportConnectionRequest, 37
_UnpackSwiAvmsEventReportDataSessionStatus, 38
_UnpackSwiAvmsEventReportHTTPStatus, 39
_UnpackSwiAvmsEventReportNotification, 40
_UnpackSwiAvmsEventReportPackageID, 41
_UnpackSwiAvmsEventReportRegStatus, 41
_UnpackSwiAvmsEventReportSessionType, 42
_UnpackSwiAvmsEventReportWAMSParamChange, 42
cat_commonEventTlv, 49
dms_ActivationStatusTlv, 56
dms_OperatingModeTlv, 57
eTWSPLMNInfoTlv, 61
loc_IPv4Info, 102
loc_IPv6Info, 102
loc_urlAddr, 109
messageModeTlv, 111
nas_PhyCaAggPcellInfo, 178
nas_PhyCaAggScellDlBw, 178
nas_PhyCaAggScellIndex, 179
nas_PhyCaAggScellIndType, 179
nas_PhyCaAggScellInfo, 182
nas_RejectReasonTlv, 190
nas_RFInfoTlv, 191
nas_SccRxInfo, 198
nas_SignalStrengthTlv, 201
nas_SLQSSignalStrengthsTlv, 204
nas_timeInfo, 212
NASBandPreferenceTlv, 225
NASEmergencyModeTlv, 225
NASGWAcqOrderPrefTlv, 226
NASLTEBandPreferenceTlv, 227
NASLteNasReleaseInfoTlv, 227
NASModePreferenceTlv, 228
NASNetSelPreferenceTlv, 228
NASOTAMessageTlv, 229
NASPhyCaAggPcellInfo, 230
NASPhyCaAggScellDlBw, 230
NASPhyCaAggScellIndex, 231
NASPhyCaAggScellIndType, 232
NASPhyCaAggScellInfo, 233
NASPRLPreferenceTlv, 233
NASRoamPreferenceTlv, 236
NASServDomainPrefTlv, 236
NASTimeInfoTlv, 238
newMTMessageTlv, 238
sMSCAddressTlv, 406
sMSEtwsMessageTlv, 407
sMSOnIMSTlv, 409
transferRouteMessageTlv, 413
unpack_pds_SetPdsState_Ind_t, 564
wds_DataULongLongTlv, 714
wds_DataULongTlv, 715
wds_DHCPLeaseOptTlv, 715
wds_DHCPLeaseStateTlv, 716
wds_DHCPProfileIdTlv, 717
wds_IPv4AdTlv, 722
Tlvresult
pack_dms_GetCustFeaturesV2_t, 241
pack_dms_SetCustFeaturesV2_t, 246
pack_dms_SetPower_t, 247
pack_dms_SetUSBComp_t, 248
pack_dms_UIMGetICCID_t, 252
pack_fms_GetImagesPreference_t, 255
pack_fms_GetStoredImages_t, 256
pack_fms_SetImagesPreference_t, 257
pack_loc_Delete_Assist_Data_t, 258
pack_loc_EventRegister_t, 260
pack_loc_SetExtPowerState_t, 260
pack_loc_SetOperationMode_t, 261
pack_loc_SLQSLOCGetBestAvailPos_t, 261
pack_loc_Start_t, 272
pack_loc_Stop_t, 272
pack_nas_SetNetworkPreference_t, 275
pack_uim_ChangePin_t, 329
pack_uim_ReadTransparent_t, 331
pack_uim_SetPinProtection_t, 332
pack_uim_UnblockPin_t, 339

pack_uim_VerifyPin_t, 340
unpack_cat_SetCATEventCallback_t, 438
unpack_dms_GetBandCapability_t, 441
unpack_dms_GetCrashAction_t, 441
unpack_dms_GetCustFeature_t, 443
unpack_dms_GetCustFeaturesV2_t, 444
unpack_dms_GetDeviceCap_t, 445
unpack_dms_GetDeviceHardwareRev_t, 447
unpack_dms_GetDeviceMfr_t, 448
unpack_dms_GetDeviceSerialNumbers_t, 449
unpack_dms_GetFirmwareInfo_t, 450
unpack_dms_GetFirmwareRevision_t, 451
unpack_dms_GetFirmwareRevisions_t, 452
unpack_dms_GetFSN_t, 453
unpack_dms_GetIMSI_t, 454
unpack_dms_GetManufacturer_t, 454
unpack_dms_GetModelID_t, 455
unpack_dms_GetNetworkTime_t, 455
unpack_dms_GetOfflineReason_t, 456
unpack_dms_GetPower_t, 457
unpack_dms_GetPRLVersion_t, 458
unpack_dms_GetUSBComp_t, 460
unpack_dms_GetVoiceNumber_t, 461
unpack_dms_ResetToFactoryDefaults_t, 462
unpack_dms_SetActivationStatusCallback_t, 462
unpack_dms_SetCustFeature_t, 463
unpack_dms_SetCustFeaturesV2_t, 463
unpack_dms_SetEventReport_ind_t, 464
unpack_dms_SetEventReport_t, 464
unpack_dms_SetFirmwarePreference_t, 464
unpack_dms_SetPower_t, 465
unpack_dms_SetUSBComp_t, 465
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 466
unpack_dms_SLQSDmsSwiGetResetInfo_t, 467
unpack_dms_SLQSDmsSwiIndicationRegister_t, 468
unpack_dms_SLQSGetERIFile_t, 471
unpack_dms_SLQSSwiClearDyingGaspStatistics_t, 472
unpack_dms_SLQSSwiGetCrashInfo_t, 472
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 473
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 473
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 476
unpack_dms_SLQSSwiGetHostDevInfo_t, 476
unpack_dms_SLQSSwiGetOSInfo_t, 477
unpack_dms_SLQSSwiGetSerialNoExt_t, 478
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 478
unpack_dms_SLQSSwiSetHostDevInfo_t, 478
unpack_dms_SLQSSwiSetOSInfo_t, 479
unpack_dms_SLQSUIMGetState_t, 479
unpack_dms_UIMGetControlKeyStatus_t, 480
unpack_dms_UIMGetICCID_t, 481
unpack_dms_UIMGetPINStatus_t, 482
unpack_dms_UIMSetControlKeyProtection_t, 483
unpack_dms_UIMSetPINProtection_t, 483
unpack_dms_UIMUnblockControlKey_t, 484
unpack_fms_GetImagesPreference_t, 485
unpack_fms_GetStoredImages_t, 485
unpack_fms_SetImagesPreference_t, 486
unpack_loc_BestAvailPos_Ind_t, 491
unpack_loc_Delete_Assist_Data_t, 492
unpack_loc_DeleteAssistData_Ind_t, 493
unpack_loc_EngineState_Ind_t, 493
unpack_loc_EventNMEA_Ind_t, 494
unpack_loc_EventRegister_t, 494
unpack_loc_GetServer_Ind_t, 495
unpack_loc_GnssSvInfo_Ind_t, 496
unpack_loc_PositionRpt_Ind_t, 502
unpack_loc_SetExtPowerConfig_Ind_t, 503
unpack_loc_SetExtPowerState_t, 504
unpack_loc_SetOperationMode_Ind_t, 504
unpack_loc_SetOperationMode_t, 505
unpack_loc_SLQSLOCGetBestAvailPos_t, 505
unpack_loc_Start_t, 505
unpack_loc_Stop_t, 506
unpack_nas_GetNetworkPreference_t, 512
unpack_nas_SetNetworkPreference_t, 520
unpack_nas_SetServingSystemCallback_ind_t, 521
unpack_nas_SlqsGetLTECphyCAInfo_t, 523
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 546
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 547
unpack_uim_ChangePin_t, 611
unpack_uim_GetCardStatus_t, 611
unpack_uim_ReadTransparent_t, 612
unpack_uim_SetPinProtection_t, 613
unpack_uim_UnblockPin_t, 618
unpack_uim_VerifyPin_t, 619
unpack_wds_SLQSCreateProfile_t, 658
unpack_wds_SLQSGetProfileSettings_t, 665
unpack_wds_SLQSSetIPFamilyPreference_t, 668
toServiceId
sms_BroadcastConfig, 393
TokenBucket
unpack_qos_swiQosFlow_t, 588
tokenRate
unpack_qos_tokenBucket_t, 588
tosMask
LibPackTFTIDParams, 93
total_rx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 577
total_rx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 577
total_tx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 577
total_tx_bytes_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 577
total_tx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 577
total_tx_pkt_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 577
TrackAreaCode

unpack_nas_SLQSGetServingSystem_t, 529
TrackingStatus
unpack_pds_SetPdsState_Ind_t, 564
TrafficClass
unpack_qos_swiQosFlow_t, 588
trafficClass
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
trafficPriority
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
TranDstPort
unpack_qos_swiQosFilter_t, 584
TranSrcPort
unpack_qos_swiQosFilter_t, 584
TransCap
sms_transLayerInfo, 405
TransType
sms_transLayerInfo, 406
transactionID
sMSTransferRouteMTMessage, 410
transferDelay
LibPackUMTSQoS, 95
wds_UMTSMinQoS, 728
TransferRouteMTMessageInfo
transferRouteMessageTlv, 413
transferRouteMessageTlv, 412
TlvPresent, 413
TransferRouteMTMessageInfo, 413
transferStatInd, 413
StatsMask, 413
StatsPeriod, 413
transferStats
pack_wds_SLQSSetWdsEventCallback_t, 378
trueSrvStatus
nas_GSMSrvStatusInfo, 139
tx_bytes
unpack_QosFlowStat_t, 590
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
tx_bytes_drp
unpack_QosFlowStat_t, 590
tx_pkt
unpack_QosFlowStat_t, 590
tx_pkt_drp
unpack_QosFlowStat_t, 590
tx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
TxDropConutTlv
unpack_RMTransferStatistics_ind_t, 591
txOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 665
TxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 591
TxOkConutTlv
unpack_RMTransferStatistics_ind_t, 591
txPower
nas_txInfo, 213
TxQFilter
unpack_qos_QosFlowInfo_t, 574
TxQFlowGranted
unpack_qos_QosFlowInfo_t, 574
type
pack_wds_GetDefaultProfileNum_t, 360
pack_wds_SetDefaultProfileNum_t, 366
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 466
unpack_dms_SLQSDmsSwiGetResetInfo_t, 467
unpack_qmi_t, 570
typeSize
unpack_sms_GetSMSCAddress_t, 593

u16PRLVersion
unpack_dms_GetPRLVersion_t, 458
u8PRLPreference
unpack_dms_GetPRLVersion_t, 458
UDPDstPort
unpack_qos_swiQosFilter_t, 584
UDPSrcPort
unpack_qos_swiQosFilter_t, 584
ULONG
SwiDataTypes.h, 984
ULONGLONG
SwiDataTypes.h, 984
UMTSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 667
UMTSInstInfo
nas_UMTSInfo, 215
UMTSLTENbrCell
nas_WCDMAInfoLTENeighborCell, 220
UMTSReqQoS
LibPackUMTSReqQoSSigInd, 96
UNIQUE_ID_LEN
dms.h, 747
UNUSEDPARAM
common.h, 737
SwiDataTypes.h, 983
uPhyCardStatus
slot_t, 390
uPhySlotStatus
slot_t, 390
USBComp
pack_dms_SetUSBComp_t, 248
unpack_dms_GetUSBComp_t, 460
USHORT
SwiDataTypes.h, 984
USSInformation
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 352
UUSData
voice_UUSInfo, 713
UUSDatalen
voice_UUSInfo, 713
UUSDcs
voice_UUSInfo, 713
UUSType
voice_UUSInfo, 713

uarfcn
nas_lteWcdmaCellInfo, 167
nas_UMTSInfo, 215
nas_wcdmaUARFCN, 224
ueInIdle
nas_LTEInfoInterfreq, 154
nas_LTEInfoIntrafreq, 156
nas_LTEInfoNeighboringGSM, 157
nas_LTEInfoNeighboringWCDMA, 158
uim.h, 995
MAX_ICCID_LENGTH, 998
MAX_NO_OF_SLOTS, 998
MAX_SLOTS_STATUS, 998
pack_uim_ChangePin, 998
pack_uim_GetCardStatus, 998
pack_uim_ReadTransparent, 999
pack_uim_SLQSUIMAuthenticate, 999
pack_uim_SLQSUIMDepersonalization, 1000
pack_uim_SLQSUIMEventRegister, 1000
pack_uim_SLQSUIMGetConfiguration, 1000
pack_uim_SLQSUIMGetFileAttributes, 1001
pack_uim_SLQSUIMGetSlotsStatus, 1001
pack_uim_SLQSUIMPowerDown, 1002
pack_uim_SLQSUIMPowerUp, 1002
pack_uim_SLQSUIMRefreshComplete, 1002
pack_uim_SLQSUIMRefreshGetLastEvent, 1003
pack_uim_SLQSUIMRefreshOK, 1003
pack_uim_SLQSUIMRefreshRegister, 1003
pack_uim_SLQSUIMReset, 1004
pack_uim_SLQSUIMSwitchSlot, 1004
pack_uim_SetPinProtection, 999
pack_uim_UnblockPin, 1004
pack_uim_VerifyPin, 1005
unpack_uim_ChangePin, 1005
unpack_uim_GetCardStatus, 1006
unpack_uim_ReadTransparent, 1006
unpack_uim_SLQSUIMAuthenticate, 1007
unpack_uim_SLQSUIMDepersonalization, 1007
unpack_uim_SLQSUIMEventRegister, 1008
unpack_uim_SLQSUIMGetConfiguration, 1008
unpack_uim_SLQSUIMGetFileAttributes, 1008
unpack_uim_SLQSUIMGetSlotsStatus, 1009
unpack_uim_SLQSUIMPowerDown, 1009
unpack_uim_SLQSUIMPowerUp, 1009
unpack_uim_SLQSUIMRefreshComplete, 1010
unpack_uim_SLQSUIMRefreshGetLastEvent, 1010
unpack_uim_SLQSUIMRefreshOK, 1010
unpack_uim_SLQSUIMRefreshRegister, 1011
unpack_uim_SLQSUIMReset, 1011
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind, 1011
unpack_uim_SLQSUIMSwitchSlot, 1012
unpack_uim_SetPinProtection, 1006
unpack_uim_SetUimSlotStatusChangeCallback_ind, 1007
unpack_uim_UnblockPin, 1012
unpack_uim_VerifyPin, 1012
uim_UIMSessionInformation, 435
aid, 436
aidLength, 436
sessionType, 436
uim_appStatus, 413
aidLength, 415
aidVal, 415
appState, 415
appType, 416
persoFeature, 416
persoRetries, 416
persoState, 416
persoUnblockRetries, 416
pin1Retries, 416
pin1State, 416
pin2Retries, 416
pin2State, 416
puk1Retries, 416
puk2Retries, 416
univPin, 416
uim_authenticateResult, 416
content, 416
contentLen, 416
uim_authenticationData, 417
context, 417
data, 417
dataLen, 417
uim_cardResult, 418
sw1, 418
sw2, 418
uim_cardStatus, 418
index1xPri, 419
index1xSec, 419
indexGwPri, 419
indexGwSec, 419
numSlot, 419
SlotInfo, 419
uim_changeUIMPIN, 419
oldPINLen, 420
oldPINVal, 420
pinID, 420
pinLen, 420
pinVal, 420
uim_depersonalizationInformation, 420
ckLen, 421
ckVal, 421
feature, 421
operation, 421
uim_encryptedPIN1, 421
pin1Len, 422
pin1Val, 422
uim_fileAttributes, 422
fileID, 425
fileSize, 425
fileType, 425
rawLen, 425
rawValue, 425
recordCount, 425

recordSize, 425
secActivate, 425
secActivateMask, 425
secDeactivate, 425
secDeactivateMask, 425
secIncrease, 425
secIncreaseMask, 425
secRead, 425
secReadMask, 425
secWrite, 425
secWriteMask, 425
uim_fileInfo, 425
fileID, 426
path, 426
pathLen, 426
uim_hotSwapStatus, 426
hotSwap, 426
hotSwapLength, 426
uim_personalizationStatus, 426
feature, 427
numFeatures, 427
unblockLeft, 427
verifyLeft, 427
uim_readResult, 427
content, 428
contentLen, 428
uim_readTransparentInfo, 428
length, 428
offset, 428
uim_refreshevent, 428
aid, 430
aidLength, 430
arrfileInfo, 430
mode, 430
numOfFiles, 430
sessionType, 430
stage, 430
uim_registerRefresh, 430
arrfileInfo, 431
numFiles, 431
registerFlag, 431
voteForInit, 431
uim_remainingRetries, 431
unblockLeft, 431
verifyLeft, 431
uim_sessionInformation, 432
aid, 432
aidLength, 432
sessionType, 432
uim_setPINProtection, 432
pinID, 433
pinLength, 433
pinOperation, 433
pinValue, 433
uim_slotInfo, 433
AppStatus, 435
cardState, 435
errorState, 435
numApp, 435
upinRetries, 435
upinState, 435
upukRetries, 435
uim_unblockUIMPIN, 436
newPINLen, 436
newPINVal, 436
pinID, 436
pukLen, 436
pukVal, 437
uim_verifyUIMPIN, 437
pinID, 437
pinLen, 437
pinVal, 437
uimSlotStatus
slots_t, 392
ulData
wds_DataULongTlv, 715
ulMask
rmTrasnferStaticsReq, 388
ulMax
_PackSwiAvmsSetSettingsPeriodInfo, 31
ulMin
_PackSwiAvmsSetSettingsPeriodInfo, 31
ulPhysicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 338
ulPkgDownloadComplete
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36
ulPkgDownloadSize
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36
ullData
wds_DataULongLongTlv, 714
umtsEcio
nas_UMTSinstInfo, 215
umtsInst
nas_UMTSInfo, 215
umtsLTENbrCellLen
nas_WCDMAInfoLTENeighborCell, 220
umtsPsc
nas_UMTSinstInfo, 215
umtsRscp
nas_UMTSinstInfo, 216
umtsUarfcn
nas_UMTSinstInfo, 216
UnPackGetProfileSettingOut, 675
curProfile, 675
pExtErrCode, 675
unblockLeft
uim_personalizationStatus, 427
uim_remainingRetries, 431
unblockRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 480
unpack_dms_UIMSetPINProtection_t, 483
unpack_dms_UIMUnblockControlKey_t, 484
uniqueID
image_info_t, 67

univPin
appStats, 46
uim_appStatus, 416
universalTime
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 543
unpack_QosFlowStat_t, 589
bearerId, 590
tx_bytes, 590
tx_bytes_drp, 590
tx_pkt, 590
tx_pkt_drp, 590
unpack_RMTransferStatistics_ind_t, 590
RxDropConutTlv, 591
RxOkByteCountTlv, 591
RxOkConutTlv, 591
TxDropConutTlv, 591
TxOkByteCountTlv, 591
TxOkConutTlv, 591
unpack_cat_CATSendEnvelopeCommand
cat.h, 733
unpack_cat_CATSendTerminalResponse
cat.h, 733
unpack_cat_SetCATEventCallback
cat.h, 734
unpack_cat_SetCATEventCallback_t, 438
errorMask, 438
Tlvresult, 438
unpack_cat_SetCatEventCallback_ind
cat.h, 734
unpack_cat_SetCatEventCallback_ind_t, 437
CCETlv, 438
event_Index, 438
unpack_dms_ActivateAutomatic
dms.h, 769
unpack_dms_GetActivationState
dms.h, 770
unpack_dms_GetActivationState_t, 438
state, 439
unpack_dms_GetBandCapability
dms.h, 770
unpack_dms_GetBandCapability_t, 439
BandCapability, 440
Tlvresult, 441
unpack_dms_GetCrashAction
dms.h, 770
unpack_dms_GetCrashAction_t, 441
DevCrashState, 441
Tlvresult, 441
unpack_dms_GetCustFeature
dms.h, 771
unpack_dms_GetCustFeature_t, 441
DHCPRelayEnabled, 443
DisableIMSI, 443
GPSLPM, 443
GPSSel, 443
GpsEnable, 443
IPFamSupport, 443
IsVoiceEnabled, 443
RMAutoConnect, 443
SMSSupport, 443
Tlvresult, 443
unpack_dms_GetCustFeaturesV2
dms.h, 771
unpack_dms_GetCustFeaturesV2_t, 443
GetCustomFeatureV2, 444
Tlvresult, 444
unpack_dms_GetDeviceCap
dms.h, 771
unpack_dms_GetDeviceCap_t, 444
DataServiceCapability, 445
MaxRXChannelRate, 445
MaxTXChannelRate, 445
RadioIfaces, 445
RadioIfacesSize, 445
SimCapability, 445
Tlvresult, 445
unpack_dms_GetDeviceCapabilities
dms.h, 772
unpack_dms_GetDeviceCapabilities_t, 445
dataServiceCaCapability, 446
maxRxChannelRate, 446
maxTxChannelRate, 446
RadioIfaces, 446
radioIfacesSize, 446
simCapability, 446
unpack_dms_GetDeviceHardwareRev
dms.h, 772
unpack_dms_GetDeviceHardwareRev_t, 446
String, 447
stringSize, 447
Tlvresult, 447
unpack_dms_GetDeviceMfr
dms.h, 772
unpack_dms_GetDeviceMfr_t, 447
String, 447
stringSize, 448
Tlvresult, 448
unpack_dms_GetDeviceSerialNumbers
dms.h, 773
unpack_dms_GetDeviceSerialNumbers_t, 448
ESNString, 449
esnSize, 449
IMEIString, 449
imeiSize, 449
imeiSvnSize, 449
ImeiSvnString, 449
MEIDString, 449
meidSize, 449
Tlvresult, 449
unpack_dms_GetFSN
dms.h, 774
unpack_dms_GetFSN_t, 452
String, 453
Tlvresult, 453
unpack_dms_GetFirmwareInfo

dms.h, 773
unpack_dms_GetFirmwareInfo_t, 449
appversion_str, 450
bootversion_str, 450
carrier_str, 450
cur_carr_name, 450
cur_carr_rev, 450
modelid_str, 450
packageid_str, 450
priversion_str, 450
sku_str, 450
Tlvresult, 450
unpack_dms_GetFirmwareRevision
dms.h, 773
unpack_dms_GetFirmwareRevision_t, 451
AMSSString, 451
amssSize, 451
PRIString, 451
Tlvresult, 451
unpack_dms_GetFirmwareRevisions
dms.h, 774
unpack_dms_GetFirmwareRevisions_t, 451
AMSSString, 452
amssSize, 452
bootSize, 452
BootString, 452
PRIString, 452
priSize, 452
Tlvresult, 452
unpack_dms_GetHardwareRevision
dms.h, 774
unpack_dms_GetHardwareRevision_t, 453
hwVer, 453
unpack_dms_GetIMSI
dms.h, 775
unpack_dms_GetIMSI_t, 453
imsi, 454
Tlvresult, 454
unpack_dms_GetManufacturer
dms.h, 775
unpack_dms_GetManufacturer_t, 454
manufacturer, 454
Tlvresult, 454
unpack_dms_GetModelID
dms.h, 775
unpack_dms_GetModelID_t, 454
modelid, 455
Tlvresult, 455
unpack_dms_GetNetworkTime
dms.h, 776
unpack_dms_GetNetworkTime_t, 455
source, 455
timestamp, 455
Tlvresult, 455
unpack_dms_GetOfflineReason
dms.h, 776
unpack_dms_GetOfflineReason_t, 456
pReasonMask, 456
pbPlatform, 456
Tlvresult, 456
unpack_dms_GetPRLVersion
dms.h, 777
unpack_dms_GetPRLVersion_t, 457
Tlvresult, 458
u16PRLVersion, 458
u8PRLPreference, 458
unpack_dms_GetPower
dms.h, 776
unpack_dms_GetPower_t, 456
HardwareControlledMode, 457
OfflineReason, 457
OperationMode, 457
Tlvresult, 457
unpack_dms_GetSerialNumbers
dms.h, 777
unpack_dms_GetSerialNumbers_t, 458
esn, 459
imei_no, 459
imeisv_svn, 459
meid, 459
unpack_dms_GetUSBComp
dms.h, 777
unpack_dms_GetUSBComp_t, 459
NumSupUSBComps, 460
SupUSBComps, 460
Tlvresult, 460
USBComp, 460
unpack_dms_GetVoiceNumber
dms.h, 778
unpack_dms_GetVoiceNumber_t, 460
MIN, 461
minSize, 461
Tlvresult, 461
VoiceNumber, 461
voiceNumberSize, 461
unpack_dms_ResetToFactoryDefaults
dms.h, 778
unpack_dms_ResetToFactoryDefaults_t, 461
Tlvresult, 462
unpack_dms_SLQSDmsSwiGetResetInfo
dms.h, 782
unpack_dms_SLQSDmsSwiGetResetInfo_Ind
dms.h, 782
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 465
source, 466
Tlvresult, 466
type, 466
unpack_dms_SLQSDmsSwiGetResetInfo_t, 466
source, 467
Tlvresult, 467
type, 467
unpack_dms_SLQSDmsSwiIndicationRegister
dms.h, 782
unpack_dms_SLQSDmsSwiIndicationRegister_t, 467
Tlvresult, 468
unpack_dms_SLQSGetBandCapability

dms.h, 783
unpack_dms_SLQSGetBandCapability_t, 468
bandCapability, 471
LteBandCapability, 471
TdsBandCapability, 471
unpack_dms_SLQSGetERIFile
dms.h, 783
unpack_dms_SLQSGetERIFile_t, 471
eriFile, 471
Tlvresult, 471
unpack_dms_SLQSSwiClearDyingGaspStatistics
dms.h, 783
unpack_dms_SLQSSwiClearDyingGaspStatistics_t, 471
Tlvresult, 472
unpack_dms_SLQSSwiGetCrashInfo
dms.h, 784
unpack_dms_SLQSSwiGetCrashInfo_t, 472
crashInfoParam, 472
Tlvresult, 472
unpack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 784
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 472
pGetDyingGaspCfg, 473
Tlvresult, 473
unpack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 784
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 473
pGetDyingGaspStatistics, 473
Tlvresult, 473
unpack_dms_SLQSSwiGetFirmwareCurr
dms.h, 785
unpack_dms_SLQSSwiGetFirmwareCurr_t, 473
carrier, 474
fwvers, 474
numEntries, 474
pCurrImgInfo, 474
pkgver, 474
priver, 474
unpack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 785
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 474
imgType, 475
logString, 475
refData, 475
refString, 475
ResCode, 476
Tlvresult, 476
unpack_dms_SLQSSwiGetHostDevInfo
dms.h, 786
unpack_dms_SLQSSwiGetHostDevInfo_t, 476
manString, 476
modelString, 476
plasmaIDString, 476
swVerString, 476
Tlvresult, 476
unpack_dms_SLQSSwiGetOSInfo
dms.h, 786
unpack_dms_SLQSSwiGetOSInfo_t, 477
nameString, 477
Tlvresult, 477
versionString, 477
unpack_dms_SLQSSwiGetSerialNoExt
dms.h, 786
unpack_dms_SLQSSwiGetSerialNoExt_t, 477
meidString, 478
Tlvresult, 478
unpack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 787
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 478
Tlvresult, 478
unpack_dms_SLQSSwiSetHostDevInfo
dms.h, 787
unpack_dms_SLQSSwiSetHostDevInfo_t, 478
Tlvresult, 478
unpack_dms_SLQSSwiSetOSInfo
dms.h, 787
unpack_dms_SLQSSwiSetOSInfo_t, 478
Tlvresult, 479
unpack_dms_SLQSUIMGetState
dms.h, 788
unpack_dms_SLQSUIMGetState_t, 479
state, 479
Tlvresult, 479
unpack_dms_SetActivationStatusCallback
dms.h, 778
unpack_dms_SetActivationStatusCallback_t, 462
Tlvresult, 462
unpack_dms_SetCrashAction
dms.h, 779
unpack_dms_SetCrashAction_t, 462
notused, 462
unpack_dms_SetCustFeature
dms.h, 779
unpack_dms_SetCustFeature_t, 462
Tlvresult, 463
unpack_dms_SetCustFeaturesV2
dms.h, 779
unpack_dms_SetCustFeaturesV2_t, 463
Tlvresult, 463
unpack_dms_SetEventReport
dms.h, 780
unpack_dms_SetEventReport_ind
dms.h, 780
unpack_dms_SetEventReport_ind_t, 463
ActivationStatusTlv, 464
OperatingModeTlv, 464
Tlvresult, 464
unpack_dms_SetEventReport_t, 464
Tlvresult, 464
unpack_dms_SetFirmwarePreference
dms.h, 780
unpack_dms_SetFirmwarePreference_t, 464
Tlvresult, 464
unpack_dms_SetPower
dms.h, 781

unpack_dms_SetPower_t, 465
Tlvresult, 465
unpack_dms_SetUSBComp
dms.h, 781
unpack_dms_SetUSBComp_t, 465
Tlvresult, 465
unpack_dms_UIMChangePIN
dms.h, 788
unpack_dms_UIMGetControlKeyStatus
dms.h, 788
unpack_dms_UIMGetControlKeyStatus_t, 479
facilityState, 480
Tlvresult, 480
unblockRetriesLeft, 480
verifyRetriesLeft, 480
unpack_dms_UIMGetICCID
dms.h, 789
unpack_dms_UIMGetICCID_t, 480
String, 481
stringSize, 481
Tlvresult, 481
unpack_dms_UIMGetPINStatus
dms.h, 789
unpack_dms_UIMGetPINStatus_t, 481
p1Status, 482
p1UnblockRetriesLeft, 482
p1VerifyRetriesLeft, 482
p2Status, 482
p2UnblockRetriesLeft, 482
p2VerifyRetriesLeft, 482
Tlvresult, 482
unpack_dms_UIMSetControlKeyProtection
dms.h, 789
unpack_dms_UIMSetControlKeyProtection_t, 482
Tlvresult, 483
verifyRetriesLeft, 483
unpack_dms_UIMSetPINProtection
dms.h, 790
unpack_dms_UIMSetPINProtection_t, 483
Tlvresult, 483
unblockRetriesLeft, 483
verifyRetriesLeft, 483
unpack_dms_UIMUnblockControlKey
dms.h, 790
unpack_dms_UIMUnblockControlKey_t, 484
Tlvresult, 484
unblockRetriesLeft, 484
unpack_dms_UIMUnblockPIN
dms.h, 790
unpack_dms_UIMVerifyPIN
dms.h, 791
unpack_dms_ValidateSPC
dms.h, 791
unpack_fms_GetImagesPreference
fms.h, 794
unpack_fms_GetImagesPreference_t, 484
ImageListSize, 485
pImageList, 485
Tlvresult, 485
unpack_fms_GetStoredImages
fms.h, 794
unpack_fms_GetStoredImages_t, 485
imageList, 485
imagelistSize, 485
Tlvresult, 485
unpack_fms_SetImagesPreference
fms.h, 794
unpack_fms_SetImagesPreference_t, 485
ImageTypes, 486
ImageTypesSize, 486
Tlvresult, 486
unpack_loc_BestAvailPos_Ind
loc.h, 812
unpack_loc_BestAvailPos_Ind_t, 486
pAltitudeWrtEllipsoid, 490
pAltitudeWrtMeanSeaLevel, 490
pGpsTime, 490
pHeading, 490
pHeadingUnc, 490
pHorCirConf, 490
pHorEllpConf, 490
pHorReliability, 490
pHorUncCircular, 491
pHorUncEllipseOrientAzimuth, 491
pHorUncEllipseSemiMajor, 491
pHorUncEllipseSemiMinor, 491
pLatitude, 491
pLongitude, 491
pMagneticDeviation, 491
pPrecisionDilution, 491
pSensorDataUsage, 491
pSpeedHorizontal, 491
pSpeedUnc, 491
pSpeedVertical, 491
pSpeedVerticalUnc, 491
pSvUsedforFix, 491
pTechnologyMask, 491
pTimeSrc, 491
pTimeUnc, 491
pTimestampUtc, 491
pVertConfidence, 491
pVertReliability, 491
pVertUnc, 491
pXid, 491
status, 491
Tlvresult, 491
unpack_loc_Delete_Assist_Data_t, 491
Tlvresult, 492
unpack_loc_DeleteAssistData
loc.h, 813
unpack_loc_DeleteAssistData_Ind
loc.h, 813
unpack_loc_DeleteAssistData_Ind_t, 492
status, 492
Tlvresult, 493
unpack_loc_EngineState_Ind

loc.h, 814
unpack_loc_EngineState_Ind_t, 493
engineState, 493
Tlvresult, 493
unpack_loc_EventNMEA_Ind
loc.h, 814
unpack_loc_EventNMEA_Ind_t, 493
NMEAData, 494
Tlvresult, 494
unpack_loc_EventRegister
loc.h, 814
unpack_loc_EventRegister_t, 494
Tlvresult, 494
unpack_loc_GetServer_Ind
loc.h, 815
unpack_loc_GetServer_Ind_t, 494
pIPv4AddrInfo, 495
pIPv6AddrInfo, 495
pURL, 495
serverStatus, 495
serverType, 495
Tlvresult, 495
unpack_loc_GnssSvInfo_Ind
loc.h, 815
unpack_loc_GnssSvInfo_Ind_t, 496
altitudeAssumed, 496
pSatelliteInfo, 496
Tlvresult, 496
unpack_loc_PositionRpt_Ind
loc.h, 815
unpack_loc_PositionRpt_Ind_t, 496
pAltitudeAssumed, 501
pAltitudeWrtEllipsoid, 501
pAltitudeWrtMeanSeaLevel, 501
pFixId, 501
pGpsTime, 501
pHeading, 501
pHeadingUnc, 502
pHorConfidence, 502
pHorReliability, 502
pHorUncCircular, 502
pHorUncEllipseOrientAzimuth, 502
pHorUncEllipseSemiMajor, 502
pHorUncEllipseSemiMinor, 502
pLatitude, 502
pLeapSeconds, 502
pLongitude, 502
pMagneticDeviation, 502
pPrecisionDilution, 502
pSensorDataUsage, 502
pSpeedHorizontal, 502
pSpeedUnc, 502
pSpeedVertical, 502
pSvUsedforFix, 502
pTechnologyMask, 502
pTimeSrc, 502
pTimeUnc, 502
pTimestampUtc, 502
pVertConfidence, 502
pVertReliability, 502
pVertUnc, 502
sessionId, 502
sessionStatus, 502
Tlvresult, 502
unpack_loc_SLQSLOCGetBestAvailPos
loc.h, 817
unpack_loc_SLQSLOCGetBestAvailPos_t, 505
Tlvresult, 505
unpack_loc_SLQSLOCGetServer
loc.h, 817
unpack_loc_SLQSLOCInjectPosition
loc.h, 818
unpack_loc_SLQSLOCInjectSensorData
loc.h, 818
unpack_loc_SLQSLOCInjectUTCTime
loc.h, 818
unpack_loc_SLQSLOCSetCradleMountConfig
loc.h, 818
unpack_loc_SetExtPowerConfig_Ind
loc.h, 816
unpack_loc_SetExtPowerConfig_Ind_t, 503
status, 503
Tlvresult, 503
unpack_loc_SetExtPowerState
loc.h, 816
unpack_loc_SetExtPowerState_t, 503
Tlvresult, 504
unpack_loc_SetOperationMode
loc.h, 816
unpack_loc_SetOperationMode_Ind
loc.h, 817
unpack_loc_SetOperationMode_Ind_t, 504
status, 504
Tlvresult, 504
unpack_loc_SetOperationMode_t, 504
Tlvresult, 505
unpack_loc_Start
loc.h, 819
unpack_loc_Start_t, 505
Tlvresult, 505
unpack_loc_Stop
loc.h, 819
unpack_loc_Stop_t, 505
Tlvresult, 506
unpack_nas_GetACCOLC
nas.h, 844
unpack_nas_GetANAAAAuthenticationStatus
nas.h, 845
unpack_nas_GetCDMANetworkParameters
nas.h, 845
unpack_nas_GetCDMANetworkParameters_t, 506
Application, 508
Broadcast, 508
CustomSCP, 508
ForceRev0, 508
Protocol, 508

RegForeignNID, 508
RegForeignSID, 508
RegHomeSID, 508
Roaming, 508
SCI, 508
SCM, 508
unpack_nas_GetHomeNetwork
nas.h, 845
unpack_nas_GetHomeNetwork3GPP2
nas.h, 846
unpack_nas_GetHomeNetwork3GPP2_t, 508
nameSize, 510
pMCC, 510
pMNC, 510
pNID, 510
pName, 510
pNw2DescDisp, 510
pNw2DescEnc, 510
pNw2DescLen, 510
pNw2MCC, 510
pNw2MNC, 510
pNw2Name, 510
pSID, 510
unpack_nas_GetHomeNetwork_t, 510
mcc, 511
mnc, 511
name, 511
nid, 511
sid, 511
unpack_nas_GetNetworkPreference
nas.h, 846
unpack_nas_GetNetworkPreference_t, 511
ActiveTechPref, 512
Duration, 512
PersistentTechPref, 512
Tlvresult, 512
unpack_nas_GetRFInfo
nas.h, 846
unpack_nas_GetRFInfo_t, 512
instancesSize, 513
RFBandInfoElements, 513
unpack_nas_GetServingNetwork
nas.h, 847
unpack_nas_GetServingNetwork_t, 513
CSDomain, 515
DataCaps, 515
DataCapsLen, 515
MCC, 515
MNC, 515
Name, 515
nameSize, 515
PSDomain, 515
RAN, 515
RadioIfaces, 515
RadioIfacesSize, 515
RegistrationState, 515
Roaming, 515
unpack_nas_GetServingNetworkCapabilities
nas.h, 847
unpack_nas_GetServingNetworkCapabilities_t, 515
DataCaps, 516
DataCapsLen, 516
unpack_nas_GetSignalStrengths
nas.h, 847
unpack_nas_GetSignalStrengths_t, 516
len, 516
radio, 516
rssi, 516
unpack_nas_InitiateDomainAttach
nas.h, 848
unpack_nas_PerformNetworkScan
nas.h, 848
unpack_nas_PerformNetworkScan_t, 517
p3GppNetworkInfoInstances, 517
p3GppNetworkInstanceSize, 518
pPCSInstance, 518
pPCSInstanceSize, 518
pRATINstance, 518
pRATInstanceSize, 518
pScanResult, 518
unpack_nas_SLQSConfigSigInfo
nas.h, 851
unpack_nas_SLQSGetErrorRate
nas.h, 852
unpack_nas_SLQSGetErrorRate_t, 521
pCDMAFrameErrRate, 522
pGSMBER, 522
pHDRPackErrRate, 522
pWCDMABER, 522
unpack_nas_SLQSGetNetworkTime
nas.h, 852
unpack_nas_SLQSGetNetworkTime_t, 523
p3GPP2TimeInfo, 523
p3GPPTimeInfo, 523
unpack_nas_SLQSGetOperatorNameData
nas.h, 853
unpack_nas_SLQSGetOperatorNameData_t, 523
pNITZInformation, 524
pOperatorNameString, 524
pOperatorPLMNList, 524
pPLMNNetworkName, 524
pSrvcProviderName, 524
unpack_nas_SLQSGetPLMNName
nas.h, 853
unpack_nas_SLQSGetPLMNName_t, 524
longName, 524
longNameCI, 524
longNameEn, 525
longNameLen, 525
longNameSB, 525
shortName, 525
shortNameCI, 525
shortNameEn, 525
shortNameLen, 525
shortNameSB, 525
spn, 525

spnEncoding, 525
spnLength, 525
unpack_nas_SLQSGetServingSystem
nas.h, 853
unpack_nas_SLQSGetServingSystem_t, 525
BasestationID, 528
BasestationLatitude, 528
BasestationLongitude, 528
CDMASystemInfoExt, 528
CallBarStatus, 528
CellID, 528
ConcSvcInfo, 528
CurrentPLMN, 528
DTMInd, 528
DataSrvCapabilities, 528
DefaultRoamInd, 528
DetailedSvcInfo, 528
Gpp2TimeZone, 528
GppNetworkDSTAdjustment, 528
GppTimeZone, 528
HdrPersonality, 528
Lac, 528
NetworkID, 528
PRLInd, 529
RoamIndicatorVal, 529
RoamingIndicatorList, 529
ServingSystem, 529
SystemID, 529
TrackAreaCode, 529
unpack_nas_SLQSGetSignalStrength
nas.h, 854
unpack_nas_SLQSGetSignalStrength_t, 529
ecioList, 531
ecioListLen, 531
errorRateList, 531
errorRateListLen, 531
Io, 531
ltersrp, 531
ltesnr, 531
rsrqInfo, 531
rxSignalStrengthList, 531
rxSignalStrengthListLen, 531
signalStrengthReqMask, 531
sinr, 531
unpack_nas_SLQSGetSysInfo
nas.h, 854
unpack_nas_SLQSGetSysInfo_t, 531
pAddCDMASysInfo, 533
pAddGSMSysInfo, 533
pAddHDRSysInfo, 533
pAddLTESysInfo, 533
pCDMASrvStatusInfo, 534
pCDMASysInfo, 534
pGSMCallBarringSysInfo, 534
pGSMCipherDomainSysInfo, 534
pGSMSrvStatusInfo, 534
pGSMSysInfo, 534
pHDRSrvStatusInfo, 534
pHDRSysInfo, 534
pLTESrvStatusInfo, 534
pLTESysInfo, 534
pLTEVoiceSupportSysInfo, 534
pWCDMASysInfo, 534
unpack_nas_SLQSGetSysSelectionPref
nas.h, 854
unpack_nas_SLQSGetSysSelectionPref_t, 534
pBandPref, 538
pEmerMode, 538
pGWAcqOrderPref, 538
pLTEBandPref, 538
pModePref, 538
pNetSelPref, 538
pPRLPref, 538
pRoamPref, 538
pSrvDomainPref, 538
unpack_nas_SLQSInitiateNetworkRegistration
nas.h, 855
unpack_nas_SLQSNASSwiGetChannelLock
nas.h, 858
unpack_nas_SLQSNASSwiGetChannelLock_t, 544
unpack_nas_SLQSNASSwiSetChannelLock
nas.h, 859
unpack_nas_SLQSNasConfigSigInfo2
nas.h, 855
unpack_nas_SLQSNasGet3GPP2Subscription
nas.h, 855
unpack_nas_SLQSNasGet3GPP2Subscription_t, 538
pDirNum, 539
unpack_nas_SLQSNasGetCellLocationInfo
nas.h, 856
unpack_nas_SLQSNasGetCellLocationInfo_t, 539
pCDMAInfo, 540
pLTEInfoInterfreq, 540
pLTEInfoIntrafreq, 540
pUMTSInfo, 540
unpack_nas_SLQSNasGetHDRColorCode
nas.h, 856
unpack_nas_SLQSNasGetHDRColorCode_t, 541
pColorCode, 541
unpack_nas_SLQSNasGetSigInfo
nas.h, 856
unpack_nas_SLQSNasGetSigInfo_t, 541
CDMASSInfo, 542
GSMSSInfo, 542
HDRSSInfo, 542
LTESSInfo, 542
unpack_nas_SLQSNasGetTxRxInfo
nas.h, 857
unpack_nas_SLQSNasGetTxRxInfo_t, 542
pRXChain0Info, 542
pRXChain1Info, 542
pTXInfo, 542
unpack_nas_SLQSNasIndicationRegisterExt
nas.h, 857
unpack_nas_SLQSNasNetworkTimeCallBack_ind
nas.h, 857

unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 542
pDayltSavAdj, 543
pRadioInterface, 543
pTimeZone, 543
universalTime, 543
unpack_nas_SLQSNasSigInfoCallback_ind
nas.h, 858
unpack_nas_SLQSNasSigInfoCallback_ind_t, 543
pRscp, 544
unpack_nas_SLQSNasSwiIndicationRegister
nas.h, 858
unpack_nas_SLQSNasSwiModemStatus
nas.h, 859
unpack_nas_SLQSNasSwiModemStatus_t, 545
commonInfo, 545
pLTEInfo, 546
unpack_nas_SLQSNasSwiOTAMessageCallback_ind
nas.h, 859
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 546
Info, 546
Tlvresult, 546
unpack_nas_SLQSNasSysInfoCallback_ind
nas.h, 860
unpack_nas_SLQSNasTimerCallback_ind
nas.h, 860
unpack_nas_SLQSNasTimerCallback_ind_t, 546
t3396_apn, 547
t3396_plmn_id, 547
t3396_val, 547
unpack_nas_SLQSSetBandPreference
nas.h, 860
unpack_nas_SLQSSetSignalStrengthsCallback
nas.h, 860
unpack_nas_SLQSSetSysSelectionPref
nas.h, 861
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind
nas.h, 861
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 547
Info, 547
Tlvresult, 547
unpack_nas_SLQSSwiGetHDRPersonality
nas.h, 861
unpack_nas_SLQSSwiGetHDRPersonality_t, 547
pCurrentPersonality, 548
pPersonalityListLength, 548
pProtocolSubtypeElement, 548
unpack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 862
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 548
pAppSubType, 548
pCurrentPrsnlty, 548
pPersonalityListLength, 548
unpack_nas_SLQSSwiGetHRPDStats
nas.h, 862
unpack_nas_SLQSSwiGetHRPDStats_t, 549
pPilotSetData, 549
pUATI, 549
unpack_nas_SLQSSwiGetLteCQI
nas.h, 862
unpack_nas_SLQSSwiGetLteCQI_t, 549
ValidityCW0, 550
ValidityCW1, 550
unpack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 863
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 550
pSccRxInfo, 550
unpack_nas_SLQSSwiNetworkDebug
nas.h, 863
unpack_nas_SLQSSwiNetworkDebug_t, 550
pDataStatusDetail, 551
pDeviceConfigDetail, 551
pNetworkStat1x, 551
pNetworkStatEVDO, 551
pObjectVer, 551
unpack_nas_SLQSSwiPSDetach
nas.h, 863
unpack_nas_SLQSSysInfoCallback_ind_t, 551
pGSMSysInfo, 554
pHDRSysInfo, 554
pLTESysInfo, 554
pSysInfoNoChange, 554
unpack_nas_SetACCOLC
nas.h, 848
unpack_nas_SetCDMANetworkParameters
nas.h, 849
unpack_nas_SetDataCapabilitiesCallback_ind
nas.h, 849
unpack_nas_SetDataCapabilitiesCallback_ind_t, 518
dataCaps, 518
dataCapsSize, 518
unpack_nas_SetEventReportInd
nas.h, 849
unpack_nas_SetEventReportInd_t, 518
RFTlv, 519
RRTlv, 519
SLQSSSTlv, 519
SSTlv, 519
unpack_nas_SetLURejectCallback
nas.h, 849
unpack_nas_SetNasLTECphyCaIndCallback_ind
nas.h, 850
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 519
unpack_nas_SetNetworkPreference
nas.h, 850
unpack_nas_SetNetworkPreference_t, 520
Tlvresult, 520
unpack_nas_SetRFInfoCallback
nas.h, 850
unpack_nas_SetRoamingIndicatorCallback_ind
nas.h, 851
unpack_nas_SetRoamingIndicatorCallback_ind_t, 520
roaming, 521
unpack_nas_SetServingSystemCallback_ind

nas.h, 851
unpack_nas_SetServingSystemCallback_ind_t, 521
SSInfo, 521
Tlvresult, 521
unpack_nas_SlqsGetLTECphyCAInfo
nas.h, 852
unpack_nas_SlqsGetLTECphyCAInfo_t, 522
LTECphyCaInfo, 523
Tlvresult, 523
unpack_omaDmConfigTlv_t, 554
alertmsg, 555
alertmsglength, 555
state, 555
userInputReq, 555
userInputTimeout, 555
unpack_omaDmFotaTlv_t, 555
description, 557
descriptionlength, 557
fwdloadsize, 557
fwloadComplete, 557
namelength, 557
package_name, 557
sessionType, 557
severity, 557
state, 557
updateCompleteStatus, 557
userInputReq, 557
userInputTimeout, 557
version, 557
versionlength, 557
unpack_omaDmNotificationsTlv_t, 557
notification, 558
sessionStatus, 558
unpack_pds_ForceXTRADownload
pds.h, 875
unpack_pds_GetPDSDefaults
pds.h, 875
unpack_pds_GetPDSDefaults_t, 558
pAccuracy, 559
pInterval, 559
pOperation, 559
pTimeout, 559
unpack_pds_GetPDSState
pds.h, 876
unpack_pds_GetPDSState_t, 559
pEnabledStatus, 559
pTrackingStatus, 559
unpack_pds_GetPortAutomaticTracking
pds.h, 876
unpack_pds_GetPortAutomaticTracking_t, 560
pbAuto, 560
unpack_pds_GetServiceAutomaticTracking
pds.h, 876
unpack_pds_GetServiceAutomaticTracking_t, 560
pbAuto, 560
unpack_pds_GetXTRAAutomaticDownload
pds.h, 877
unpack_pds_GetXTRAAutomaticDownload_t, 560
pInterval, 561
pbEnabled, 561
unpack_pds_GetXTRANetwork
pds.h, 877
unpack_pds_GetXTRANetwork_t, 561
pPreference, 561
unpack_pds_GetXTRAValidity
pds.h, 877
unpack_pds_GetXTRAValidity_t, 561
pDuration, 562
pGPSWeek, 562
pGPSWeekOffset, 562
unpack_pds_PDSInjectTimeReference
pds.h, 878
unpack_pds_ResetPDSData
pds.h, 878
unpack_pds_SLQSGetAGPSConfig
pds.h, 881
unpack_pds_SLQSGetAGPSConfig_t, 564
pServerAddress, 565
pServerPort, 565
pServerURL, 565
unpack_pds_SLQSGetGPSStateInfo
pds.h, 882
unpack_pds_SLQSGetGPSStateInfo_t, 565
Altitude, 568
EngineState, 568
HorizontalUncertainty, 569
Iono_valid, 569
Latitude, 569
Longitude, 569
Time_uncert_ms, 569
ValidMask, 569
VerticalUncertainty, 569
unpack_pds_SLQSPDSDeterminePosition
pds.h, 882
unpack_pds_SLQSPDSInjectAbsoluteTimeReference
pds.h, 882
unpack_pds_SLQSPDSInjectPositionData
pds.h, 883
unpack_pds_SLQSSetAGPSConfig
pds.h, 883
unpack_pds_SLQSSetPositionMethodState
pds.h, 883
unpack_pds_SetEventReport_Ind
pds.h, 878
unpack_pds_SetEventReport_Ind_t, 562
dLatitude, 563
dLongitude, 563
has_PositionDataNMEA, 564
has_SessionStatus, 564
has_dLatitude, 563
has_dLongitude, 563
has_posSrc, 564
posSrc, 564
PositionDataNMEA, 564
SessionStatus, 564
unpack_pds_SetEventReportCallback

pds.h, 879
unpack_pds_SetPDSDefaults
pds.h, 879
unpack_pds_SetPDSState
pds.h, 879
unpack_pds_SetPdsState_Ind
pds.h, 880
unpack_pds_SetPdsState_Ind_t, 564
EnabledStatus, 564
TlvPresent, 564
TrackingStatus, 564
unpack_pds_SetPortAutomaticTracking
pds.h, 880
unpack_pds_SetServiceAutomaticTracking
pds.h, 880
unpack_pds_SetXTRAAutomaticDownload
pds.h, 881
unpack_pds_SetXTRANetwork
pds.h, 881
unpack_pds_StartPDSTrackingSessionExt
pds.h, 884
unpack_pds_StopPDSTrackingSession
pds.h, 884
unpack_qmi_t, 569
msgid, 570
type, 570
xid, 570
unpack_qos_IPv4Addr_t, 570
addr, 570
subnetMask, 571
unpack_qos_IPv6Addr_t, 571
addr, 571
prefixLen, 571
unpack_qos_IPv6TrafCls_t, 571
mask, 571
val, 572
unpack_qos_Port_t, 572
port, 572
range, 572
unpack_qos_QosFlowInfo_t, 572
BearerID, 574
is_RxQFlowGranted_Available, 574
is_TxQFlowGranted_Available, 574
NumRxFilters, 574
NumTxFilters, 574
QFlowState, 574
RxQFilter, 574
RxQFlowGranted, 574
TxQFilter, 574
TxQFlowGranted, 574
unpack_qos_QosFlowInfoState_t, 574
id, 574
isNewFlow, 574
state, 575
unpack_qos_SLQSQosGetNetworkStatus
qos.h, 935
unpack_qos_SLQSQosGetNetworkStatus_t, 575
NWQoSStatus, 575
unpack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 936
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 575
ambr_dl, 576
ambr_dl_ext, 576
ambr_dl_ext2, 576
ambr_ul, 576
ambr_ul_ext, 576
ambr_ul_ext2, 576
apnId, 576
unpack_qos_SLQSQosSwiReadDataStats
qos.h, 936
unpack_qos_SLQSQosSwiReadDataStats_t, 576
apnId, 577
numQosFlow, 577
qosFlow, 577
total_rx_bytes, 577
total_rx_pkt, 577
total_tx_bytes, 577
total_tx_pkt, 577
unpack_qos_SLQSSetQosEventCallback
qos.h, 937
unpack_qos_SLQSSetQosEventCallback_ind
qos.h, 937
unpack_qos_SLQSSetQosEventCallback_ind_t, 578
NumFlows, 578
QosFlowInfo, 578
unpack_qos_SLQSSetQosNWStatusCallback_ind
qos.h, 937
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 578
status, 578
unpack_qos_SLQSSetQosPriEventCallback_ind
qos.h, 938
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 579
event, 579
unpack_qos_SLQSSetQosStatusCallback_ind
qos.h, 939
unpack_qos_SLQSSetQosStatusCallback_ind_t, 579
event, 580
id, 580
reason, 580
status, 580
unpack_qos_Tos_t, 588
mask, 589
val, 589
unpack_qos_dataRate_t, 570
dataRateMax, 570
guaranteedRate, 570
unpack_qos_pktErrRate_t, 572
exponent, 572
multiplier, 572
unpack_qos_swiQosFilter_t, 580
EspSpi, 582
IPv4DstAddr, 583
IPv4SrcAddr, 583
IPv4Tos, 583

IPv6DstAddr, 583
IPv6Label, 583
IPv6SrcAddr, 583
IPv6TrafCls, 583
Id, 582
index, 583
is_EspSpi_Available, 583
is_IPv4DstAddr_Available, 583
is_IPv4SrcAddr_Available, 583
is_IPv4Tos_Available, 583
is_IPv6DstAddr_Available, 583
is_IPv6Label_Available, 583
is_IPv6SrcAddr_Available, 583
is_IPv6TrafCls_Available, 583
is_Id_Available, 583
is_NxtHdrProto_Available, 583
is_Precedence_Available, 583
is_TCPDstPort_Available, 583
is_TCPSrcPort_Available, 583
is_TranDstPort_Available, 583
is_TranSrcPort_Available, 583
is_UDPDstPort_Available, 583
is_UDPSrcPort_Available, 583
NxtHdrProto, 583
Precedence, 583
TCPDstPort, 583
TCPSrcPort, 584
TranDstPort, 584
TranSrcPort, 584
UDPDstPort, 584
UDPSrcPort, 584
version, 584
unpack_qos_swiQosFlow_t, 584
DataRate, 587
index, 587
is_DataRate_Available, 587
is_Jitter_Available, 587
is_Latency_Available, 587
is_LteQci_Available, 587
is_MaxAllowedPktSz_Available, 587
is_MinPolicedPktSz_Available, 587
is_PktErrRate_Available, 587
is_ProfileId3GPP2_Available, 587
is_TokenBucket_Available, 587
is_TrafficClass_Available, 587
is_val_3GPP2Pri_Available, 587
is_val_3GPPImCn_Available, 587
is_val_3GPPSigInd_Available, 587
Jitter, 587
Latency, 587
LteQci, 587
MaxAllowedPktSz, 587
MinPolicedPktSz, 587
PktErrRate, 588
ProfileId3GPP2, 588
TokenBucket, 588
TrafficClass, 588
val_3GPP2Pri, 588
val_3GPPImCn, 588
val_3GPPResResidualBER, 588
val_3GPPSigInd, 588
val_3GPPTraHdlPri, 588
unpack_qos_tokenBucket_t, 588
bucketSz, 588
peakRate, 588
tokenRate, 588
unpack_result_code_only
common.h, 739
unpack_rms_GetSMSWake
rms.h, 940
unpack_rms_GetSMSWake_t, 590
enabled, 590
wake_mask, 590
unpack_rms_SetSMSWake
rms.h, 941
unpack_sar_SLQSGetRfSarState
sar.h, 942
unpack_sar_SLQSGetRfSarState_t, 591
pSarRFState, 592
unpack_sar_SLQSSetRfSarState
sar.h, 942
unpack_sms_GetSMSCAddress
sms.h, 957
unpack_sms_GetSMSCAddress_t, 592
addressSize, 592
pSMSCAddress, 592
pSMSCType, 593
typeSize, 593
unpack_sms_SLQSDeleteSMS
sms.h, 959
unpack_sms_SLQSDeleteSMS_t, 595
unpack_sms_SLQSGetIndicationRegister
sms.h, 959
unpack_sms_SLQSGetIndicationRegister_t, 595
pGetIndicationRegInfo, 595
unpack_sms_SLQSGetMessageWaiting
sms.h, 959
unpack_sms_SLQSGetMessageWaiting_t, 595
pGetMsgWaitingInfoResp, 595
unpack_sms_SLQSGetSMS
sms.h, 960
unpack_sms_SLQSGetSMS_t, 596
message, 596
messageFormat, 596
messageSize, 596
messageTag, 596
unpack_sms_SLQSGetSMSList
sms.h, 960
unpack_sms_SLQSGetSMSList_t, 597
messageList, 597
messageListSize, 597
unpack_sms_SLQSGetSmsBroadcastConfig
sms.h, 960
unpack_sms_SLQSGetSmsBroadcastConfig_t, 596
pBroadcastConfig, 597
pCDMABroadcastConfig, 597

unpack_sms_SLQSGetTransLayerInfo
sms.h, 961
unpack_sms_SLQSGetTransLayerInfo_t, 597
pGetTransLayerInfo, 598
unpack_sms_SLQSGetTransNWRegInfo
sms.h, 961
unpack_sms_SLQSGetTransNWRegInfo_t, 598
unpack_sms_SLQSModifySMSStatus
sms.h, 961
unpack_sms_SLQSModifySMSStatus_t, 598
unpack_sms_SLQSSendAsyncSMS
sms.h, 962
unpack_sms_SLQSSetIndicationRegister
sms.h, 962
unpack_sms_SLQSSetSmsBroadcastActivation
sms.h, 962
unpack_sms_SLQSSetSmsBroadcastConfig
sms.h, 963
unpack_sms_SLQSSetSmsStorage
sms.h, 963
unpack_sms_SLQSSmsGetMaxStorageSize
sms.h, 963
unpack_sms_SLQSSmsGetMaxStorageSize_t, 598
pMaxStorageSizeResp, 599
unpack_sms_SLQSSmsGetMessageProtocol
sms.h, 964
unpack_sms_SLQSSmsGetMessageProtocol_t, 599
pMessageProtocol, 599
unpack_sms_SLQSSmsSetRoutes
sms.h, 964
unpack_sms_SLQSSwiGetSMSStorage
sms.h, 964
unpack_sms_SLQSSwiGetSMSStorage_t, 599
pSmsStorage, 600
unpack_sms_SLQSWmsMemoryFullCallBack_ind
sms.h, 965
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 600
messageMode, 600
storageType, 600
unpack_sms_SaveSMS
sms.h, 957
unpack_sms_SaveSMS_t, 593
pMessageIndex, 593
unpack_sms_SendSMS
sms.h, 957
unpack_sms_SendSMS_t, 593
messageFailureCode, 593
messageID, 593
unpack_sms_SetNewSMSCallback
sms.h, 958
unpack_sms_SetNewSMSCallback_ind
sms.h, 958
unpack_sms_SetNewSMSCallback_ind_t, 593
ETWSTlv, 594
IMSTlv, 594
MMTlv, 594
NewMMTlv, 594
SMSCTlv, 594
TRMessageTlv, 595
unpack_sms_SetNewSMSCallback_t, 595
unpack_sms_SetSMSCAddress
sms.h, 958
unpack_swiavms_SLQSAVMSEventReportInd
swiavms.h, 979
unpack_swiavms_SLQSAVMSEventReportInd_t
swiavms.h, 968
unpack_swiavms_SLQSAVMSGetSettings
swiavms.h, 979
unpack_swiavms_SLQSAVMSGetSettings_t, 600
AutoConnect, 601
AutoReboot, 601
FwAutoSDM, 601
FwAutoUpdate, 601
FwAutodownload, 601
pAPNInfo, 601
pConnectionRetryTimers, 601
pNotificationStore, 601
pPeroidsInfo, 601
pPollingTimer, 601
resultcode, 601
unpack_swiavms_SLQSAVMSSendSelection
swiavms.h, 980
unpack_swiavms_SLQSAVMSSendSelection_t, 601
resultcode, 601
unpack_swiavms_SLQSAVMSSessionGetInfo
swiavms.h, 980
unpack_swiavms_SLQSAVMSSessionGetInfo_t
swiavms.h, 969
unpack_swiavms_SLQSAVMSSetSettings
swiavms.h, 981
unpack_swiavms_SLQSAVMSSetSettings_t, 602
resultcode, 602
unpack_swiavms_SLQSAVMSStartSession
swiavms.h, 982
unpack_swiavms_SLQSAVMSStartSession_t, 602
resultcode, 602
sessionResponse, 603
unpack_swiavms_SLQSAVMSStopSession
swiavms.h, 982
unpack_swiavms_SLQSAVMSStopSession_t, 603
resultcode, 603
unpack_swiavms_SLQSAvmsSetEventReport
swiavms.h, 981
unpack_swiavms_SLQSAvmsSetEventReport_t, 601
resultcode, 602
unpack_swiloc_SwiLocGetAutoStart
swiloc.h, 985
unpack_swiloc_SwiLocGetAutoStart_t, 603
fix_rate, 604
fix_rate_reported, 604
fix_type, 604
fix_type_reported, 604
function, 604
function_reported, 604
max_dist, 604

max_dist_reported, 604
max_time, 605
max_time_reported, 605
unpack_swiloc_SwiLocSetAutoStart
swiloc.h, 985
unpack_swioma_SLQSOMADMAlertCallback
swioma.h, 991
unpack_swioma_SLQSOMADMAlertCallback_ind
swioma.h, 992
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 605
eventType, 605
unpack_swioma_SLQSOMADMCancelSession
swioma.h, 992
unpack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 993
unpack_swioma_SLQSOMADMGetSessionInfo_t, 605
Date, 607
DateLength, 607
PkgDescLength, 607
PkgDescription, 607
PkgName, 608
PkgNameLength, 608
RetryCount, 608
SessionState, 608
SessionType, 608
Severity, 608
Source, 608
SourceLength, 608
Status, 608
Time, 608
TimeLength, 608
UpdateCompleteStatus, 608
unpack_swioma_SLQSOMADMGetSettings
swioma.h, 993
unpack_swioma_SLQSOMADMGetSettings_t, 608
Autosdm, 609
FOTAdownload, 609
FwAutoCheck, 609
unpack_swioma_SLQSOMADMSendSelection
swioma.h, 993
unpack_swioma_SLQSOMADMSetSettings
swioma.h, 994
unpack_swioma_SLQSOMADMStartSession
swioma.h, 994
unpack_swioma_SLQSOMADMStartSession_t, 610
FwAvailability, 610
unpack_uim_ChangePin
uim.h, 1005
unpack_uim_ChangePin_t, 610
pEncryptedPIN1, 611
pIndicationToken, 611
pRemainingRetries, 611
Tlvresult, 611
unpack_uim_GetCardStatus
uim.h, 1006
unpack_uim_GetCardStatus_t, 611
pCardStatus, 611
pHotSwapStatus, 611
Tlvresult, 611
unpack_uim_ReadTransparent
uim.h, 1006
unpack_uim_ReadTransparent_t, 611
pCardResult, 612
pEncryptedData, 612
pIndicationToken, 612
pReadResult, 612
Tlvresult, 612
unpack_uim_SLQSUIMAuthenticate
uim.h, 1007
unpack_uim_SLQSUIMAuthenticate_t, 614
pAuthenticateResult, 614
pCardResult, 614
pIndicationToken, 614
unpack_uim_SLQSUIMDepersonalization
uim.h, 1007
unpack_uim_SLQSUIMDepersonalization_t, 614
pRemainingRetries, 615
unpack_uim_SLQSUIMEventRegister
uim.h, 1008
unpack_uim_SLQSUIMEventRegister_t, 615
eventMask, 615
unpack_uim_SLQSUIMGetConfiguration
uim.h, 1008
unpack_uim_SLQSUIMGetConfiguration_t, 615
pAutoSelection, 616
pHaltSubscription, 616
pPersonalizationStatus, 616
unpack_uim_SLQSUIMGetFileAttributes
uim.h, 1008
unpack_uim_SLQSUIMGetFileAttributes_t, 616
pCardResult, 616
pFileAttributes, 616
pIndicationToken, 616
unpack_uim_SLQSUIMGetSlotsStatus
uim.h, 1009
unpack_uim_SLQSUIMGetSlotsStatus_t, 617
pNumberOfPhySlot, 617
pUimSlotsStatus, 617
unpack_uim_SLQSUIMPowerDown
uim.h, 1009
unpack_uim_SLQSUIMPowerUp
uim.h, 1009
unpack_uim_SLQSUIMRefreshComplete
uim.h, 1010
unpack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 1010
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 617
pRefreshEvent, 617
unpack_uim_SLQSUIMRefreshOK
uim.h, 1010
unpack_uim_SLQSUIMRefreshRegister
uim.h, 1011
unpack_uim_SLQSUIMReset
uim.h, 1011
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind

uim.h, 1011
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t, 617
unpack_uim_SLQSUIMSwitchSlot
uim.h, 1012
unpack_uim_SetPinProtection
uim.h, 1006
unpack_uim_SetPinProtection_t, 612
pEncryptedPIN1, 613
pIndicationToken, 613
pRemainingRetries, 613
Tlvresult, 613
unpack_uim_SetUimSlotStatusChangeCallback_ind
uim.h, 1007
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 613
bNumberOfPhySlots, 613
slotsstatusChange, 613
unpack_uim_UnblockPin
uim.h, 1012
unpack_uim_UnblockPin_t, 618
pEncryptedPIN1, 618
pIndicationToken, 618
pRemainingRetries, 618
Tlvresult, 618
unpack_uim_VerifyPin
uim.h, 1012
unpack_uim_VerifyPin_t, 618
pEncryptedPIN1, 619
pIndicationToken, 619
pRemainingRetries, 619
Tlvresult, 619
unpack_voice_AnswerUSSD
voice.h, 1029
unpack_voice_CancelUSSD
voice.h, 1030
unpack_voice_OriginateUSSD
voice.h, 1030
unpack_voice_SLQSOriginateUSSD
voice.h, 1030
unpack_voice_SLQSOriginateUSSD_t, 619
pAlphaIDInfo, 619
pCCSuppsType, 620
pCallId, 619
pCcResultType, 620
pUSSDInfo, 620
pfailureCause, 620
unpack_voice_SLQSVoiceALSSelectLine
voice.h, 1031
unpack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 1031
unpack_voice_SLQSVoiceAnswerCall
voice.h, 1031
unpack_voice_SLQSVoiceAnswerCall_t, 620
pCallId, 620
unpack_voice_SLQSVoiceBindSubscription
voice.h, 1032
unpack_voice_SLQSVoiceBurstDTMF
voice.h, 1032
unpack_voice_SLQSVoiceBurstDTMF_t, 620
pCallID, 620
unpack_voice_SLQSVoiceDialCall
voice.h, 1032
unpack_voice_SLQSVoiceDialCall_t, 621
pAlphaIDInfo, 621
pCCResultType, 621
pCCSUPSType, 621
pCallID, 621
unpack_voice_SLQSVoiceEndCall
voice.h, 1033
unpack_voice_SLQSVoiceEndCall_t, 621
pCallId, 622
unpack_voice_SLQSVoiceGetAllCallInfo
voice.h, 1033
unpack_voice_SLQSVoiceGetAllCallInfo_t, 622
pArrAlertingPattern, 624
pArrAlertingType, 624
pArrAlphaID, 624
pArrCallEndReason, 624
pArrCallInfo, 624
pArrCalledPartyNum, 624
pArrConnectPartyNum, 624
pArrDiagInfo, 624
pArrRedirPartyNum, 624
pArrRemotePartyName, 624
pArrRemotePartyNum, 624
pArrSvcOption, 624
pArrUUSInfo, 624
pVoicePrivacy, 624
unpack_voice_SLQSVoiceGetCLIP
voice.h, 1035
unpack_voice_SLQSVoiceGetCLIP_t, 631
pAlphaIDInfo, 632
pCCResType, 632
pCallID, 632
pFailCause, 632
unpack_voice_SLQSVoiceGetCLIR
voice.h, 1035
unpack_voice_SLQSVoiceGetCLIR_t, 632
pAlphaIDInfo, 633
pCCResType, 633
pCallID, 633
pFailCause, 633
unpack_voice_SLQSVoiceGetCNAP
voice.h, 1035
unpack_voice_SLQSVoiceGetCNAP_t, 633
pAlphaIDInfo, 634
pCCResType, 634
pCallID, 634
pFailCause, 634
unpack_voice_SLQSVoiceGetCOLP
voice.h, 1036
unpack_voice_SLQSVoiceGetCOLP_t, 634
pAlphaIDInfo, 635
pCCResType, 636
pCallID, 635

pFailCause, 636
unpack_voice_SLQSVoiceGetCOLR
voice.h, 1036
unpack_voice_SLQSVoiceGetCOLR_t, 636
pAlphaIDInfo, 637
pCCResType, 637
pCallID, 637
pFailCause, 637
unpack_voice_SLQSVoiceGetCallBarring
voice.h, 1033
unpack_voice_SLQSVoiceGetCallBarring_t, 624
pAlphaIDInfo, 625
pCCResType, 625
pCallID, 625
pFailCause, 625
pSvcClass, 625
unpack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 1034
unpack_voice_SLQSVoiceGetCallForwardingStatus_t, 626
pAlphaIDInfo, 627
pCCResType, 627
pCallID, 627
pFailCause, 627
unpack_voice_SLQSVoiceGetCallInfo
voice.h, 1034
unpack_voice_SLQSVoiceGetCallInfo_t, 627
pAlertType, 629
pAlertingPattern, 629
pAlphaIDInfo, 629
pCallInfo, 629
pConnectNumInfo, 629
pDiagInfo, 629
pOTASPStatus, 629
pRemotePartyName, 629
pRemotePartyNum, 629
pSrvOpt, 629
pUUSInfo, 629
pVoicePrivacy, 629
unpack_voice_SLQSVoiceGetCallWaiting
voice.h, 1034
unpack_voice_SLQSVoiceGetCallWaiting_t, 629
pAlphaIDInfo, 630
pCCResType, 631
pCallID, 631
pFailCause, 631
pSvcClass, 631
unpack_voice_SLQSVoiceGetConfig
voice.h, 1036
unpack_voice_SLQSVoiceGetConfig_t, 637
pAirTimerCnt, 638
pAutoAnswerStat, 639
pCurAMRConfig, 639
pCurPrefVoiceSO, 639
pCurVoiceDomainPref, 639
pCurVoicePrivacyPref, 639
pCurrTTYMode, 639
pRoamTimerCnt, 639
unpack_voice_SLQSVoiceIndicationRegister
voice.h, 1037
unpack_voice_SLQSVoiceManageCalls
voice.h, 1037
unpack_voice_SLQSVoiceManageCalls_t, 639
pFailCause, 639
unpack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 1037
unpack_voice_SLQSVoiceSendFlash
voice.h, 1038
unpack_voice_SLQSVoiceSendFlash_t, 639
pCallID, 640
unpack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 1038
unpack_voice_SLQSVoiceSetCallBarringPassword_t, 640
pAlphaIDInfo, 640
pCCResType, 641
pCallID, 641
pFailCause, 641
unpack_voice_SLQSVoiceSetConfig
voice.h, 1038
unpack_voice_SLQSVoiceSetConfig_t, 641
pAirTimerStatus, 642
pAutoAnsStatus, 642
pPrefVoiceSOStatus, 642
pRoamTimerStatus, 642
pTTYConfigStatus, 642
pVoiceDomainPrefStatus, 642
unpack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 1039
unpack_voice_SLQSVoiceSetSUPSService
voice.h, 1039
unpack_voice_SLQSVoiceSetSUPSService_t, 642
pCallID, 643
pFailCause, 643
unpack_voice_SLQSVoiceStartContDTMF
voice.h, 1039
unpack_voice_SLQSVoiceStartContDTMF_t, 643
pCallID, 644
unpack_voice_SLQSVoiceStopContDTMF
voice.h, 1040
unpack_voice_SLQSVoiceStopContDTMF_t, 644
callID, 644
unpack_wds_DHCPv4ClientLease_ind
wds.h, 1065
unpack_wds_DHCPv4ClientLease_ind_t, 644
DHCPv4LeaseOptTlv, 645
DHCPv4LeaseStateTlv, 645
IPv4AddrTlv, 645
ProfileIdTlv, 645
unpack_wds_DHCPv4ClientLeaseChange
wds.h, 1066
unpack_wds_GetAutoconnect
wds.h, 1066
unpack_wds_GetAutoconnect_t, 645
psetting, 645
unpack_wds_GetByteTotals

wds.h, 1066
unpack_wds_GetByteTotals_t, 645
pRXTotalBytes, 646
pTXTotalBytes, 646
unpack_wds_GetConnectionRate
wds.h, 1067
unpack_wds_GetConnectionRate_t, 646
currentChannelRXRate, 646
currentChannelTXRate, 646
maxChannelRXRate, 646
maxChannelTXRate, 646
unpack_wds_GetDataBearerTechnology
wds.h, 1067
unpack_wds_GetDataBearerTechnology_t, 647
pDataBearer, 647
unpack_wds_GetDefaultProfile
wds.h, 1067
unpack_wds_GetDefaultProfile_t, 647
apnname, 649
apnsize, 649
auth, 649
ipaddr, 649
ipaddrv6, 649
name, 649
namesize, 649
pdptype, 649
pridns, 649
pridnsv6, 649
secdns, 649
secdnsv6, 649
username, 649
usersize, 649
unpack_wds_GetDefaultProfileNum
wds.h, 1068
unpack_wds_GetDefaultProfileNum_t, 649
index, 650
unpack_wds_GetDormancyState
wds.h, 1068
unpack_wds_GetDormancyState_t, 650
dormancyState, 650
unpack_wds_GetLastMobileIPError
wds.h, 1068
unpack_wds_GetLastMobileIPError_t, 650
error, 650
unpack_wds_GetMobileIP
wds.h, 1069
unpack_wds_GetMobileIP_t, 651
mipMode, 651
unpack_wds_GetMobileIPProfile
wds.h, 1069
unpack_wds_GetMobileIPProfile_t, 651
AAASPI, 652
AAAState, 652
address, 652
enabled, 652
HASPI, 652
HAState, 653
NAI, 653
naiSize, 653
primaryHA, 653
revTunneling, 653
secondaryHA, 653
unpack_wds_GetPacketStatistics
wds.h, 1069
unpack_wds_GetPacketStatistics_t, 653
pRXDroppedCount, 654
pRXOKBytesLastCall, 654
pRXOkBytesCount, 654
pRXPacketErrors, 654
pRXPacketOverflows, 654
pRXPacketSuccesses, 654
pTXDroppedCount, 654
pTXOKBytesLastCall, 654
pTXOkBytesCount, 654
pTXPacketErrors, 654
pTXPacketOverflows, 654
pTXPacketSuccesses, 654
unpack_wds_GetPacketStatus
wds.h, 1070
unpack_wds_GetPacketStatus_t, 654
rXDroppedCount, 656
rXOKBytesLastCall, 656
rXOkBytesCount, 656
rXPacketErrors, 656
rXPacketOverflows, 656
rXPacketSuccesses, 656
tXDroppedCount, 656
tXOKBytesLastCall, 656
tXOkBytesCount, 656
tXPacketErrors, 656
tXPacketOverflows, 656
tXPacketSuccesses, 656
unpack_wds_GetSessionDuration
wds.h, 1070
unpack_wds_GetSessionDuration_t, 656
callDuration, 656
unpack_wds_GetSessionState
wds.h, 1071
unpack_wds_GetSessionState_t, 657
connectionStatus, 657
unpack_wds_RMSetTransferStatistics
wds.h, 1071
unpack_wds_RMSetTransferStatistics_t, 657
unpack_wds_RMTransferStatistics_ind
wds.h, 1071
unpack_wds_SLQSCreateProfile
wds.h, 1074
unpack_wds_SLQSCreateProfile_t, 657
pCreateProfileOut, 658
pProfileID, 658
Tlvresult, 658
unpack_wds_SLQSDeleteProfile
wds.h, 1074
unpack_wds_SLQSDeleteProfile_t, 658
extendedErrorCode, 658
unpack_wds_SLQSGet3GPPConfigItem

wds.h, 1074
unpack_wds_SLQSGet3GPPConfigItem_t, 658
_3gppRelease, 659
defaultPDNEnabled, 659
LTEAttachProfile, 659
profileList, 660
unpack_wds_SLQSGetCurrDataSystemStat
wds.h, 1075
unpack_wds_SLQSGetCurrDataSystemStat_t, 660
currNetworkInfo, 660
networkInfoLen, 660
prefNetwork, 660
unpack_wds_SLQSGetCurrentChannelRate
wds.h, 1075
unpack_wds_SLQSGetCurrentChannelRate_t, 660
max_channel_rx_rate, 661
max_channel_tx_rate, 661
unpack_wds_SLQSGetDUNCallInfo
wds.h, 1076
unpack_wds_SLQSGetDUNCallInfo_t, 662
callEndReason, 664
channelRate, 664
connectionStatus, 664
dataBearerTech, 664
dormancyStatus, 664
lastCallDataBearerTech, 664
mdmCallDurationActive, 664
rxOKBytesCount, 664
txOKBytesCount, 665
unpack_wds_SLQSGetDataBearerTechnology
wds.h, 1075
unpack_wds_SLQSGetDataBearerTechnology_t, 661
curDataBearerTechnology, 662
dataBearerMask, 662
lastCallDataBearerTechnology, 662
unpack_wds_SLQSGetProfileSettings
wds.h, 1076
unpack_wds_SLQSGetProfileSettings_t, 665
pProfileSettings, 665
ProfileType, 665
Tlvresult, 665
unpack_wds_SLQSGetRuntimeSettings
wds.h, 1076
unpack_wds_SLQSGetRuntimeSettings_t, 665
APNName, 666
Authentication, 667
DomainList, 667
GPRSGrantedQoS, 667
GWAddressV4, 667
IMCNflag, 667
IPFamilyPreference, 667
IPV6AddrInfo, 667
IPV6GWAddrInfo, 667
IPv4, 667
Mtu, 667
PDPType, 667
PrimaryDNSV4, 667
PrimaryDNSV6, 667
ProfileID, 667
ProfileName, 667
SecondaryDNSV4, 667
SecondaryDNSV6, 667
ServerAddrList, 667
SubnetMaskV4, 667
Technology, 667
UMTSGrantedQoS, 667
Username, 667
unpack_wds_SLQSModifyProfile
wds.h, 1077
unpack_wds_SLQSModifyProfile_t, 667
pExtErrorCode, 668
unpack_wds_SLQSResetPacketStatics
wds.h, 1077
unpack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 1079
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 671
pHwConfig, 671
unpack_wds_SLQSSGetLoopback
wds.h, 1079
unpack_wds_SLQSSGetLoopback_t, 672
ByteLoopbackMode, 672
ByteLoopbackMultiplier, 672
unpack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 1080
unpack_wds_SLQSSSetLoopback
wds.h, 1080
unpack_wds_SLQSSet3GPPConfigItem
wds.h, 1077
unpack_wds_SLQSSetIPFamilyPreference
wds.h, 1077
unpack_wds_SLQSSetIPFamilyPreference_t, 668
Tlvresult, 668
unpack_wds_SLQSSetPacketSrvStatusCallback
wds.h, 1078
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 668
bearerID, 669
conn_status, 669
ipFamily, 669
reconfigReqd, 669
sessionEndReason, 669
techName, 669
verboseSessnEndReason, 669
verboseSessnEndReasonType, 669
unpack_wds_SLQSSetWdsEventCallback
wds.h, 1078
unpack_wds_SLQSSetWdsEventCallback_ind
wds.h, 1078
unpack_wds_SLQSSetWdsEventCallback_ind_t, 669
currDBTechAvail, 670
currNWInfo, 670
dBTechAvail, 670
dBTechnology, 670
dBtechExtAvail, 670
dBtechnologyExt, 670
dataSysStatAvail, 670
dormancyStatAvail, 671

dormancyStatus, 671
mipStatus, 671
mipstatAvail, 671
netInfoLen, 671
prefNetwork, 671
ratMask, 671
rx_bytes, 671
rx_pkts, 671
soMask, 671
tx_bytes, 671
tx_pkts, 671
xferStatAvail, 671
unpack_wds_SLQSStartDataSession
wds.h, 1080
unpack_wds_SLQSStartDataSession_t, 672
pFailureReason, 673
pVerboseFailReasonType, 673
pVerboseFailureReason, 673
psid, 673
unpack_wds_SLQSStopDataSession
wds.h, 1081
unpack_wds_SLQSWdsGoActive
wds.h, 1081
unpack_wds_SLQSWdsGoDormant
wds.h, 1081
unpack_wds_SLQSWdsSetEventReport
wds.h, 1081
unpack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 1082
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 673
apnName, 674
bearerId, 674
contextId, 674
ipv4Address, 674
ipv4GWAddress, 674
ipv6Address, 675
ipv6GWAddress, 675
unpack_wds_SetAutoconnect
wds.h, 1072
unpack_wds_SetDefaultProfile
wds.h, 1072
unpack_wds_SetDefaultProfileNum
wds.h, 1072
unpack_wds_SetMobileIP
wds.h, 1073
unpack_wds_SetMobileIPParameters
wds.h, 1073
unpack_wds_SetMobileIPProfile
wds.h, 1073
unpack_wds_SetMobileIPProfile_t, 657
UnpackQmiProfileInfo
wds.h, 1046
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo
swiavms.h, 969
UnpackSwiAvmsEventReportConfig
swiavms.h, 970
UnpackSwiAvmsEventReportConnectionRequest
swiavms.h, 970
UnpackSwiAvmsEventReportDataSessionStatus
swiavms.h, 971
UnpackSwiAvmsEventReportHTTPStatus
swiavms.h, 971
UnpackSwiAvmsEventReportNotification
swiavms.h, 972
UnpackSwiAvmsEventReportPackageID
swiavms.h, 973
UnpackSwiAvmsEventReportRegStatus
swiavms.h, 973
UnpackSwiAvmsEventReportSessionType
swiavms.h, 973
UnpackSwiAvmsEventReportWAMSParamChange
swiavms.h, 974
unpackWdsProfileParam, 675
SlqsProfile3GPP, 676
SlqsProfile3GPP2, 676
UpdateCompleteStatus
unpack_swioma_SLQSOMADMGetSessionInfo_t, 608
updateCompleteStatus
unpack_omaDmFotaTlv_t, 557
upinRetries
slotInf, 392
uim_slotInfo, 435
upinState
slotInf, 392
uim_slotInfo, 435
upukRetries
slotInf, 392
uim_slotInfo, 435
usageMask
loc_sensorDataUsage, 107
userInputReq
unpack_omaDmConfigTlv_t, 555
unpack_omaDmFotaTlv_t, 557
userInputTimeout
unpack_omaDmConfigTlv_t, 555
unpack_omaDmFotaTlv_t, 557
Username
unpack_wds_SLQSGetRuntimeSettings_t, 667
username
unpack_wds_GetDefaultProfile_t, 649
usersize
unpack_wds_GetDefaultProfile_t, 649
ussDCS
pack_voice_SLQSOriginateUSSD_t, 342
voice_USSInfo, 712
ussData
pack_voice_SLQSOriginateUSSD_t, 341
voice_USSInfo, 711
ussLen
pack_voice_SLQSOriginateUSSD_t, 342
voice_USSInfo, 712
uusInfo
voice_allCallsUUSInfo, 678

VDOP

loc_precisionDilution, 104
val
unpack_qos_IPv6TrafCls_t, 572
unpack_qos_Tos_t, 589
val_3GPP2Pri
unpack_qos_swiQosFlow_t, 588
val_3GPPImCn
unpack_qos_swiQosFlow_t, 588
val_3GPPResResidualBER
unpack_qos_swiQosFlow_t, 588
val_3GPPSigInd
unpack_qos_swiQosFlow_t, 588
val_3GPPTraHdlPri
unpack_qos_swiQosFlow_t, 588
ValidMask
unpack_pds_SLQSGetGPSStateInfo_t, 569
validMask
loc_satelliteInfo, 106
ValidityCW0
unpack_nas_SLQSSwiGetLteCQI_t, 550
ValidityCW1
unpack_nas_SLQSSwiGetLteCQI_t, 550
value
pack_dms_UIMSetPINProtection_t, 253
pack_dms_UIMVerifyPIN_t, 255
value_length
DMScustSettingInfo, 58
pack_dms_SetCustFeaturesV2_t, 246
verboseSessnEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 669
verboseSessnEndReasonType
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 669
verifyLeft
uim_personalizationStatus, 427
uim_remainingRetries, 431
verifyPIN
pack_uim_VerifyPin_t, 340
verifyRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 480
unpack_dms_UIMSetControlKeyProtection_t, 483
unpack_dms_UIMSetPINProtection_t, 483
version
unpack_omaDmFotaTlv_t, 557
unpack_qos_swiQosFilter_t, 584
versionString
pack_dms_SLQSSwiSetOSInfo_t, 250
unpack_dms_SLQSSwiGetOSInfo_t, 477
versionlength
unpack_omaDmFotaTlv_t, 557
vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 267
vertReliability
pack_loc_SLQSLOCInjectPosition_t, 267
vertUnc
pack_loc_SLQSLOCInjectPosition_t, 267
VerticalUncertainty
unpack_pds_SLQSGetGPSStateInfo_t, 569
VirtStream
nas_protocolSubtypeElement, 187
voice.h
LITE_VOICE_SUPS_SRV_CLASS_DATA, 1017
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITASYNC, 1018
LITE_VOICE_SUPS_SRV_CLASS_DATACIRCUITSYNC, 1018
LITE_VOICE_SUPS_SRV_CLASS_FAX, 1018
LITE_VOICE_SUPS_SRV_CLASS_NONE, 1017
LITE_VOICE_SUPS_SRV_CLASS_PACKETACCESS, 1018
LITE_VOICE_SUPS_SRV_CLASS_PADACCESS, 1018
LITE_VOICE_SUPS_SRV_CLASS_SMS, 1018
LITE_VOICE_SUPS_SRV_CLASS_VOICE, 1017
voice.h, 1013
liteServiceClassInformation, 1017
MAXVOICEUSSDLENGTH, 1017
pack_voice_AnswerUSSD, 1018
pack_voice_CancelUSSD, 1018
pack_voice_OriginateUSSD, 1018
pack_voice_SLQSOriginateUSSD, 1019
pack_voice_SLQSVoiceALSSelectLine, 1019
pack_voice_SLQSVoiceALSSetLineSwitching, 1019
pack_voice_SLQSVoiceAnswerCall, 1020
pack_voice_SLQSVoiceBindSubscription, 1020
pack_voice_SLQSVoiceBurstDTMF, 1021
pack_voice_SLQSVoiceDialCall, 1021
pack_voice_SLQSVoiceEndCall, 1021
pack_voice_SLQSVoiceGetAllCallInfo, 1022
pack_voice_SLQSVoiceGetCLIP, 1024
pack_voice_SLQSVoiceGetCLIR, 1024
pack_voice_SLQSVoiceGetCNAP, 1024
pack_voice_SLQSVoiceGetCOLP, 1025
pack_voice_SLQSVoiceGetCOLR, 1025
pack_voice_SLQSVoiceGetCallBarring, 1022
pack_voice_SLQSVoiceGetCallForwardingStatus, 1022
pack_voice_SLQSVoiceGetCallInfo, 1023
pack_voice_SLQSVoiceGetCallWaiting, 1023
pack_voice_SLQSVoiceGetConfig, 1025
pack_voice_SLQSVoiceIndicationRegister, 1026
pack_voice_SLQSVoiceManageCalls, 1026
pack_voice_SLQSVoiceOrigUSSDNoWait, 1026
pack_voice_SLQSVoiceSendFlash, 1027
pack_voice_SLQSVoiceSetCallBarringPassword, 1027
pack_voice_SLQSVoiceSetConfig, 1027
pack_voice_SLQSVoiceSetPreferredPrivacy, 1028
pack_voice_SLQSVoiceSetSUPSService, 1028
pack_voice_SLQSVoiceStartContDTMF, 1029
pack_voice_SLQSVoiceStopContDTMF, 1029
unpack_voice_AnswerUSSD, 1029
unpack_voice_CancelUSSD, 1030
unpack_voice_OriginateUSSD, 1030

unpack_voice_SLQSOriginateUSSD, 1030
unpack_voice_SLQSVoiceALSSelectLine, 1031
unpack_voice_SLQSVoiceALSSetLineSwitching, 1031
unpack_voice_SLQSVoiceAnswerCall, 1031
unpack_voice_SLQSVoiceBindSubscription, 1032
unpack_voice_SLQSVoiceBurstDTMF, 1032
unpack_voice_SLQSVoiceDialCall, 1032
unpack_voice_SLQSVoiceEndCall, 1033
unpack_voice_SLQSVoiceGetAllCallInfo, 1033
unpack_voice_SLQSVoiceGetCLIP, 1035
unpack_voice_SLQSVoiceGetCLIR, 1035
unpack_voice_SLQSVoiceGetCNAP, 1035
unpack_voice_SLQSVoiceGetCOLP, 1036
unpack_voice_SLQSVoiceGetCOLR, 1036
unpack_voice_SLQSVoiceGetCallBarring, 1033
unpack_voice_SLQSVoiceGetCallForwarding-Status, 1034
unpack_voice_SLQSVoiceGetCallInfo, 1034
unpack_voice_SLQSVoiceGetCallWaiting, 1034
unpack_voice_SLQSVoiceGetConfig, 1036
unpack_voice_SLQSVoiceIndicationRegister, 1037
unpack_voice_SLQSVoiceManageCalls, 1037
unpack_voice_SLQSVoiceOrigUSSDNoWait, 1037
unpack_voice_SLQSVoiceSendFlash, 1038
unpack_voice_SLQSVoiceSetCallBarringPassword, 1038
unpack_voice_SLQSVoiceSetConfig, 1038
unpack_voice_SLQSVoiceSetPreferredPrivacy, 1039
unpack_voice_SLQSVoiceSetSUPSService, 1039
unpack_voice_SLQSVoiceStartContDTMF, 1039
unpack_voice_SLQSVoiceStopContDTMF, 1040
voice_CLIPResp, 694
ActiveStatus, 695
ProvisionStatus, 695
voice_CLIRResp, 695
ActiveStatus, 696
ProvisionStatus, 696
voice_CNAPResp, 696
ActiveStatus, 696
ProvisionStatus, 696
voice_COLPResp, 697
ActiveStatus, 697
ProvisionStatus, 697
voice_COLRResp, 697
ActiveStatus, 698
ProvisionStatus, 698
voice_CUGInfo, 700
CUGIndex, 700
SuppOA, 700
SuppPrefCUG, 700
voice_DTMFLengths, 702
DTMFInterdigitInterval, 702
DTMFPulseWidth, 702
voice_USSInfo, 711
ussDCS, 712
ussData, 711
ussLen, 712
voice_UUSInfo, 712
UUSData, 713
UUSDatalen, 713
UUSDcs, 713
UUSType, 713
voice_airTimer, 676
airTimerValue, 676
namID, 676
voice_allCallsAlphaIDInfo, 677
AlphaIDInfo, 677
callID, 677
voice_allCallsDiagInfo, 677
callID, 677
DiagInfo, 677
voice_allCallsUUSInfo, 678
callID, 678
uusInfo, 678
voice_alphaIDInfo, 678
alphaDcs, 679
alphaLen, 679
alphaText, 679
voice_arrAlertingPattern, 679
alertingPattern, 679
callID, 679
numInstances, 679
voice_arrAlertingType, 680
AlertingType, 680
callID, 680
numInstances, 680
voice_arrAlphaID, 680
allCallsAlphaIDInfoArr, 681
numInstances, 681
voice_arrCallEndReason, 681
callEndReason, 682
callID, 682
numInstances, 682
voice_arrCallInfo, 682
getAllCallInfo, 682
numInstances, 682
voice_arrCalledPartyNum, 681
CalledPartyNum, 681
numInstances, 681
voice_arrConnectPartyNum, 683
ConnectedPartyNum, 683
numInstances, 683
voice_arrDiagInfo, 683
DiagInfo, 684
numInstances, 684
voice_arrRedirPartyNum, 684
numInstances, 684
RedirPartyNum, 684
voice_arrRemotePartyName, 684
GetAllCallRmtPtyName, 685
numInstances, 685
voice_arrRemotePartyNum, 685
numInstances, 685
RmtPtyNum, 685

voice_arrSvcOption, 685
callID, 686
numInstances, 686
srvOption, 686
voice_arrUUSInfo, 686
AllCallsUUSInfo, 686
numInstances, 686
voice_burstDTMFInfo, 686
digitCnt, 687
pCallID, 687
pDigitBuff, 687
voice_callFWExtInfo, 689
noReplyTimer, 691
numLen, 691
numPlan, 691
numType, 691
number, 691
PI, 691
SI, 691
SvcClass, 691
SvcStatus, 691
voice_callFWInfo, 691
noReplyTimer, 692
numLen, 692
number, 692
SvcClass, 692
SvcStatus, 692
voice_callFwdTypeAndPlan, 688
numberPlan, 689
numberType, 689
voice_callInfo, 692
callID, 693
callState, 693
callType, 694
direction, 694
mode, 694
voice_calledPartySubAdd, 687
extBit, 688
oddEvenInd, 688
subAddr, 688
subAddrLen, 688
subAddrType, 688
voice_ccSUPSType, 694
reason, 694
svcType, 694
voice_connectNumInfo, 698
callerID, 699
callerIDLen, 699
numPlan, 699
numPresInd, 699
numType, 700
screeningInd, 700
voice_curAMRConfig, 700
gsmAmrStat, 701
wcdmaAmrStat, 701
voice_diagInfo, 701
diagInfoLen, 701
diagnosticInfo, 701
voice_getAllCallInformation, 702
ALS, 703
Callinfo, 703
isEmpty, 703
voice_getAllCallRmtPtyName, 703
callID, 703
RemotePartyName, 703
voice_getAllCallRmtPtyNum, 703
callID, 704
RemotePartyNum, 704
voice_getCallFWExtInfo, 704
CallFWExtInfo, 704
numInstances, 704
voice_getCallFWInfo, 705
CallFWInfo, 705
numInstances, 705
voice_peerNumberInfo, 705
callID, 707
numLen, 707
numPI, 707
numPlan, 707
numSI, 707
numType, 707
number, 707
voice_prefVoiceSO, 707
evrcCapability, 709
homeOrigVoiceSO, 709
homePageVoiceSO, 709
namID, 709
roamOrigVoiceSO, 709
voice_remotePartyName, 709
callerName, 709
codingScheme, 710
nameLen, 710
namePI, 710
voice_remotePartyNum, 710
numLen, 710
presentationInd, 710
remPartyNumber, 710
voice_roamTimer, 710
namID, 711
roamTimerValue, 711
VoiceNumber
unpack_dms_GetVoiceNumber_t, 461
voiceNumberSize
unpack_dms_GetVoiceNumber_t, 461
voiceSvc
pack_voice_SLQSVoiceSetSUPSService_t, 358
voteForInit
uim_registerRefresh, 431

wAlertMsgLength
_UnpackSwiAvmsEventReportConfig, 37
WCDMACellInfo
nas_lteWcdmaCellInfo, 167
WCDMAECIOThreshListLen
nas_WCDMAECIOThresh, 219
WCDMARSSIThreshListLen
nas_WCDMARSSIThresh, 221

WCDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 542
wDescriptionLength
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36
wErrorCode
_UnpackSwiAvmsEventReportDataSessionStatus, 38
wHTTPStatus
_UnpackSwiAvmsEventReportHTTPStatus, 39
wNameLength
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36
WORD
SwiDataTypes.h, 984
wSessionStatus
_UnpackSwiAvmsEventReportNotification, 41
wUpdateCompeteStatus
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36
wUserInputTimeout
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36
_UnpackSwiAvmsEventReportConfig, 37
_UnpackSwiAvmsEventReportConnectionRequest, 37
wVersionLength
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 36
wWamsChangeMask
_UnpackSwiAvmsEventReportWAMSParam-Change, 42
wake_mask
pack_rms_SetSMSWake_t, 311
unpack_rms_GetSMSWake_t, 590
wcdmaAmrStat
voice_curAMRConfig, 701
wcdmaRRCState
nas_WCDMAInfoLTENeighborCell, 220
wds.h
QMI_LITE_WDS_CURRENT_CALL_DB_MASK, 1046
QMI_LITE_WDS_LAST_CALL_DB_MASK, 1046
wds.h, 1040
BYT_STAT_STAT_MASK, 1046
liteQmiDataBearerMasks, 1046
PACK_WDS_IPV4, 1046
PACK_WDS_IPV6, 1046
pack_wds_DHCPv4ClientLeaseChange, 1046
pack_wds_GetAutoconnect, 1047
pack_wds_GetByteTotals, 1047
pack_wds_GetConnectionRate, 1048
pack_wds_GetDataBearerTechnology, 1048
pack_wds_GetDefaultProfile, 1048
pack_wds_GetDefaultProfileNum, 1049
pack_wds_GetDormancyState, 1049
pack_wds_GetLastMobileIPError, 1050
pack_wds_GetMobileIP, 1050
pack_wds_GetMobileIPProfile, 1050
pack_wds_GetPacketStatistics, 1051
pack_wds_GetPacketStatus, 1051
pack_wds_GetSessionDuration, 1052
pack_wds_GetSessionState, 1052
pack_wds_RMSetTransferStatistics, 1053
pack_wds_SLQSCreateProfile, 1055
pack_wds_SLQSDeleteProfile, 1056
pack_wds_SLQSGet3GPPConfigItem, 1056
pack_wds_SLQSGetCurrDataSystemStat, 1057
pack_wds_SLQSGetCurrentChannelRate, 1057
pack_wds_SLQSGetDUNCallInfo, 1058
pack_wds_SLQSGetDataBearerTechnology, 1057
pack_wds_SLQSGetProfileSettings, 1058
pack_wds_SLQSGetRuntimeSettings, 1059
pack_wds_SLQSModifyProfile, 1059
pack_wds_SLQSResetPacketStatics, 1060
pack_wds_SLQSSGetDHCPv4ClientConfig, 1061
pack_wds_SLQSSGetLoopback, 1062
pack_wds_SLQSSSetDHCPv4ClientConfig, 1062
pack_wds_SLQSSSetLoopback, 1062
pack_wds_SLQSSet3GPPConfigItem, 1060
pack_wds_SLQSSetIPFamilyPreference, 1060
pack_wds_SLQSSetWdsEventCallback, 1061
pack_wds_SLQSStartDataSession, 1063
pack_wds_SLQSStopDataSession, 1063
pack_wds_SLQSWdsGoActive, 1064
pack_wds_SLQSWdsGoDormant, 1064
pack_wds_SLQSWdsSetEventReport, 1064
pack_wds_SLQSWdsSwiPDPRuntimeSettings, 1065
pack_wds_SetAutoconnect, 1053
pack_wds_SetDefaultProfile, 1053
pack_wds_SetDefaultProfileNum, 1054
pack_wds_SetMobileIP, 1054
pack_wds_SetMobileIPParameters, 1054
pack_wds_SetMobileIPProfile, 1055
unpack_wds_DHCPv4ClientLease_ind, 1065
unpack_wds_DHCPv4ClientLeaseChange, 1066
unpack_wds_GetAutoconnect, 1066
unpack_wds_GetByteTotals, 1066
unpack_wds_GetConnectionRate, 1067
unpack_wds_GetDataBearerTechnology, 1067
unpack_wds_GetDefaultProfile, 1067
unpack_wds_GetDefaultProfileNum, 1068
unpack_wds_GetDormancyState, 1068
unpack_wds_GetLastMobileIPError, 1068
unpack_wds_GetMobileIP, 1069
unpack_wds_GetMobileIPProfile, 1069
unpack_wds_GetPacketStatistics, 1069
unpack_wds_GetPacketStatus, 1070
unpack_wds_GetSessionDuration, 1070
unpack_wds_GetSessionState, 1071
unpack_wds_RMSetTransferStatistics, 1071
unpack_wds_RMTransferStatistics_ind, 1071
unpack_wds_SLQSCreateProfile, 1074
unpack_wds_SLQSDeleteProfile, 1074
unpack_wds_SLQSGet3GPPConfigItem, 1074

unpack_wds_SLQSGetCurrDataSystemStat, 1075
unpack_wds_SLQSGetCurrentChannelRate, 1075
unpack_wds_SLQSGetDUNCallInfo, 1076
unpack_wds_SLQSGetDataBearerTechnology, 1075
unpack_wds_SLQSGetProfileSettings, 1076
unpack_wds_SLQSGetRuntimeSettings, 1076
unpack_wds_SLQSModifyProfile, 1077
unpack_wds_SLQSResetPacketStatics, 1077
unpack_wds_SLQSSGetDHCPv4ClientConfig, 1079
unpack_wds_SLQSSGetLoopback, 1079
unpack_wds_SLQSSSetDHCPv4ClientConfig, 1080
unpack_wds_SLQSSSetLoopback, 1080
unpack_wds_SLQSSet3GPPConfigItem, 1077
unpack_wds_SLQSSetIPFamilyPreference, 1077
unpack_wds_SLQSSetPacketSrvStatusCallback, 1078
unpack_wds_SLQSSetWdsEventCallback, 1078
unpack_wds_SLQSSetWdsEventCallback_ind, 1078
unpack_wds_SLQSStartDataSession, 1080
unpack_wds_SLQSStopDataSession, 1081
unpack_wds_SLQSWdsGoActive, 1081
unpack_wds_SLQSWdsGoDormant, 1081
unpack_wds_SLQSWdsSetEventReport, 1081
unpack_wds_SLQSWdsSwiPDPRuntimeSettings, 1082
unpack_wds_SetAutoconnect, 1072
unpack_wds_SetDefaultProfile, 1072
unpack_wds_SetDefaultProfileNum, 1072
unpack_wds_SetMobileIP, 1073
unpack_wds_SetMobileIPParameters, 1073
unpack_wds_SetMobileIPProfile, 1073
UnpackQmiProfileInfo, 1046
wds_DHCPLeaseOptTlv, 715
numOpt, 715
optList, 715
optListData, 715
TlvPresent, 715
wds_DHCPLeaseStateTlv, 716
leaseState, 716
TlvPresent, 716
wds_DHCPOpt, 716
optCode, 717
optValLen, 717
pOptVal, 717
wds_DHCPProfileIdTlv, 717
profileId, 717
profileType, 717
TlvPresent, 717
wds_DHCPv4HWConfig, 717
chaddr, 718
chaddrLen, 718
hwType, 718
wds_DHCPv4Option, 718
optCode, 718
optVal, 718
optValLen, 718
wds_DHCPv4OptionList, 719
numOpt, 719
pOptList, 719
wds_DHCPv4ProfileId, 719
profileId, 719
profileType, 719
wds_DataULongLongTlv, 714
TlvPresent, 714
ullData, 714
wds_DataULongTlv, 715
TlvPresent, 715
ulData, 715
wds_Domain, 720
domainLen, 720
domainName, 720
wds_DomainNameList, 720
domain, 720
numInstances, 720
wds_GPRSQoS, 721
delayClass, 721
meanThroughputClass, 721
peakThroughputClass, 721
precedenceClass, 721
reliabilityClass, 721
wds_IPV6AddressInfo, 722
IPAddressV6, 722
IPV6PrefixLen, 722
wds_IPV6GWAddressInfo, 722
gwAddressV6, 723
gwV6PrefixLen, 723
wds_IPv4AdTlv, 721
IPv4Addr, 722
TlvPresent, 722
wds_PCSCFFQDNAddress, 723
fqdnAddr, 723
fqdnLen, 723
wds_PCSCFFQDNAddressList, 723
numInstances, 724
pcsfFQDNAddress, 724
wds_PCSCFIPv4ServerAddressList, 724
numInstances, 724
pscsfIPv4Addr, 724
wds_ProfileIdentifier, 724
profileIndex, 725
profileType, 725
wds_TrStatInd, 725
statsMask, 726
statsPeriod, 726
wds_UMTSMinQoS, 726
deliveryErrSDU, 728
grntDownlinkBitrate, 728
grntUplinkBitrate, 728
maxDownlinkBitrate, 728
maxSDUSize, 728
maxUplinkBitrate, 728
qosDeliveryOrder, 728

resBerRatio, 728
sduErrorRatio, 728
trafficClass, 728
trafficPriority, 728
transferDelay, 728
wds_currNetworkInfo, 713
NetworkType, 714
RATMask, 714
SOMask, 714
wds_profileInfo, 725
SlqsProfile3GPP, 725
SlqsProfile3GPP2, 725
wdsDhcpv4HwConfig, 728
chaddr, 729
chaddrLen, 729
hwType, 729
wdsDhcpv4Option, 729
optCode, 729
optVal, 729
optValLen, 729
wdsDhcpv4OptionList, 730
numOpt, 730
pOptList, 730
wdsDhcpv4ProfileId, 730
profileId, 730
profileType, 730

xAxis
sensorData_t, 389
xferStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 671
xid
pack_loc_SLQSLOCGetBestAvailPos_t, 261
pack_qmi_t, 309
unpack_qmi_t, 570
xtra_start_gps_minutes
unpack_pds_SLQSGetGPSStateInfo_t, 569
xtra_start_gps_week
unpack_pds_SLQSGetGPSStateInfo_t, 569
xtra_valid_duration_hours
unpack_pds_SLQSGetGPSStateInfo_t, 569

yAxis
sensorData_t, 389
year
nas_timeInfo, 212
nas_UniversalTime, 218

zAxis
sensorData_t, 389