LinuxQMISDK-Lite

SLQS04.00.05

Generated by Doxygen 1.8.6

Wed Mar 8 2017 07:18:31

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 21

5.1 File List . . . 21

6 Module Documentation 23

6.1 QMI pack/unpack (lite-qmi) . . . 23
6.1.1 Detailed Description . . . 23
6.2 Streaming Download Protocol (lite-fw) . . . 24
6.2.1 Detailed Description . . . 24

7 Namespace Documentation 25

7.1 Tables Namespace Reference . . . 25
7.1.1 Detailed Description . . . 25

8 Data Structure Documentation 27

8.1 _litefw_FirmwareInfo_ Struct Reference . . . 27
8.1.1 Detailed Description . . . 27
8.1.2 Field Documentation . . . 27

8.1.2.1 szCarrier_str . . . 28
8.1.2.2 szCarrierPriversion_str . . . 28
8.1.2.3 szFwversion_str . . . 28
8.1.2.4 szModelid_str . . . 28
8.1.2.5 szPackageid_str . . . 28
8.1.2.6 szSku_str . . . 28
8.2 _PackSwiAvmsSetSettingsAPNInfo Struct Reference . . . 28
8.2.1 Detailed Description . . . 28
8.2.2 Field Documentation . . . 28
8.2.2.1 bAPNLength . . . 28
8.2.2.2 bPWDLength . . . 28
8.2.2.3 bUnameLength . . . 28
8.2.2.4 szAPN . . . 28
8.2.2.5 szPWD . . . 28
8.2.2.6 szUname . . . 28
8.3 _PackSwiAvmsSetSettingsConnectionRetryTimers Struct Reference . . . 28
8.3.1 Field Documentation . . . 29
8.3.1.1 Timers . . . 29
8.4 _PackSwiAvmsSetSettingsPeriodInfo Struct Reference . . . 29
8.4.1 Detailed Description . . . 29
8.4.2 Field Documentation . . . 29
8.4.2.1 ulMax . . . 29
8.4.2.2 ulMin . . . 29
8.5 _PackSwiAVMSSettingsAPNInfo Struct Reference . . . 29
8.5.1 Field Documentation . . . 29
8.5.1.1 bAPNLength . . . 29
8.5.1.2 bPWDLength . . . 29
8.5.1.3 bUnameLength . . . 29
8.5.1.4 pAPN . . . 29
8.5.1.5 pPWD . . . 29
8.5.1.6 pUname . . . 29
8.6 _PackSwiAVMSSettingsConnectionRetryTimers Struct Reference . . . 29
8.6.1 Field Documentation . . . 30
8.6.1.1 Timers . . . 30
8.7 _PackSwiAVMSSettingsPeriodsInfo Struct Reference . . . 30
8.7.1 Field Documentation . . . 30
8.7.1.1 max . . . 30
8.7.1.2 min . . . 30
8.8 _unpack_swiavms_SLQSAvMSEventReportInd_t Struct Reference . . . 30
8.8.1 Detailed Description . . . 30

8.8.2 Field Documentation . . . 31
8.8.2.1 pBinaryUpdateSessionInfo . . . 31
8.8.2.2 pConfig . . . 31
8.8.2.3 pConnectionRequest . . . 31
8.8.2.4 pDataSessionStatus . . . 31
8.8.2.5 pHTTPStatus . . . 31
8.8.2.6 pNotification . . . 31
8.8.2.7 pPackageID . . . 31
8.8.2.8 pRegStatus . . . 31
8.8.2.9 pSessionType . . . 31
8.8.2.10 pWAMSParaChanged . . . 31
8.8.2.11 resultcode . . . 31
8.9 _unpack_swiavms_SLQSAvMSSessionGetInfo_t Struct Reference . . . 31
8.9.1 Detailed Description . . . 32
8.9.2 Field Documentation . . . 32
8.9.2.1 pBinaryUpdateSessionInfo . . . 32
8.9.2.2 pConfig . . . 32
8.9.2.3 pNotification . . . 32
8.9.2.4 pPackageID . . . 32
8.9.2.5 resultcode . . . 32
8.10 _UnpackSwiAvmsEventReportBinaryUpdateSessionInfo Struct Reference . . . 32
8.10.1 Detailed Description . . . 32
8.10.2 Field Documentation . . . 33
8.10.2.1 bBinaryType . . . 33
8.10.2.2 bSerity . . . 33
8.10.2.3 bState . . . 33
8.10.2.4 bUserInputRequest . . . 33
8.10.2.5 szDescription . . . 33
8.10.2.6 szName . . . 34
8.10.2.7 szVersion . . . 34
8.10.2.8 TlvPresent . . . 34
8.10.2.9 ulPkgDownloadComplete . . . 34
8.10.2.10 ulPkgDownloadSize . . . 34
8.10.2.11 wDescriptionLength . . . 34
8.10.2.12 wNameLength . . . 34
8.10.2.13 wUpdateCompeteStatus . . . 34
8.10.2.14 wUserInputTimeout . . . 34
8.10.2.15 wVersionLength . . . 34
8.11 _UnpackSwiAvmsEventReportConfig Struct Reference . . . 34
8.11.1 Detailed Description . . . 34

8.11.2 Field Documentation . . . 34
8.11.2.1 bState . . . 35
8.11.2.2 bUserInputRequest . . . 35
8.11.2.3 szAlertMsg . . . 35
8.11.2.4 TlvPresent . . . 35
8.11.2.5 wAlertMsgLength . . . 35
8.11.2.6 wUserInputTimeout . . . 35
8.12 _UnpackSwiAvmsEventReportConnectionRequest Struct Reference . . . 35
8.12.1 Detailed Description . . . 35
8.12.2 Field Documentation . . . 35
8.12.2.1 bUserInputRequest . . . 35
8.12.2.2 TlvPresent . . . 35
8.12.2.3 wUserInputTimeout . . . 35
8.13 _UnpackSwiAvmsEventReportDataSessionStatus Struct Reference . . . 35
8.13.1 Detailed Description . . . 35
8.13.2 Field Documentation . . . 36
8.13.2.1 bType . . . 36
8.13.2.2 TlvPresent . . . 36
8.13.2.3 wErrorCode . . . 36
8.14 _UnpackSwiAvmsEventReportHTTPStatus Struct Reference . . . 36
8.14.1 Detailed Description . . . 36
8.14.2 Field Documentation . . . 37
8.14.2.1 TlvPresent . . . 37
8.14.2.2 wHTTPStatus . . . 38
8.15 _UnpackSwiAvmsEventReportNotification Struct Reference . . . 38
8.15.1 Detailed Description . . . 38
8.15.2 Field Documentation . . . 38
8.15.2.1 bNotification . . . 38
8.15.2.2 TlvPresent . . . 39
8.15.2.3 wSessionStatus . . . 39
8.16 _UnpackSwiAvmsEventReportPackageID Struct Reference . . . 39
8.16.1 Detailed Description . . . 39
8.16.2 Field Documentation . . . 39
8.16.2.1 bPackageID . . . 39
8.16.2.2 TlvPresent . . . 39
8.17 _UnpackSwiAvmsEventReportRegStatus Struct Reference . . . 39
8.17.1 Detailed Description . . . 39
8.17.2 Field Documentation . . . 39
8.17.2.1 bRegStatus . . . 39
8.17.2.2 TlvPresent . . . 39

8.18 _UnpackSwiAvmsEventReportSessionType Struct Reference . . . 39
8.18.1 Detailed Description . . . 40
8.18.2 Field Documentation . . . 40
8.18.2.1 bType . . . 40
8.18.2.2 TlvPresent . . . 40
8.19 _UnpackSwiAvmsEventReportWAMSParamChange Struct Reference . . . 40
8.19.1 Detailed Description . . . 40
8.19.2 Field Documentation . . . 40
8.19.2.1 TlvPresent . . . 40
8.19.2.2 wWamsChangeMask . . . 40
8.20 altSrcInfo_t Struct Reference . . . 40
8.20.1 Detailed Description . . . 41
8.20.2 Field Documentation . . . 41
8.20.2.1 coverage . . . 41
8.20.2.2 linkage . . . 41
8.20.2.3 source . . . 41
8.21 appStats Struct Reference . . . 41
8.21.1 Detailed Description . . . 42
8.21.2 Field Documentation . . . 44
8.21.2.1 aidLength . . . 44
8.21.2.2 aidVal . . . 44
8.21.2.3 appState . . . 44
8.21.2.4 appType . . . 44
8.21.2.5 persoFeature . . . 44
8.21.2.6 persoRetries . . . 44
8.21.2.7 persoState . . . 44
8.21.2.8 persoUnblockRetries . . . 44
8.21.2.9 pin1Retries . . . 44
8.21.2.10 pin1State . . . 44
8.21.2.11 pin2Retries . . . 44
8.21.2.12 pin2State . . . 44
8.21.2.13 puk1Retries . . . 44
8.21.2.14 puk2Retries . . . 44
8.21.2.15 univPin . . . 44
8.22 CarrierImage_t Struct Reference . . . 44
8.22.1 Detailed Description . . . 45
8.22.2 Field Documentation . . . 45
8.22.2.1 m_FwBuildId . . . 45
8.22.2.2 m_FwImageId . . . 45
8.22.2.3 m_nCarrierId . . . 45

8.22.2.4 m_nFolderId . . . 45
8.22.2.5 m_nStorage . . . 45
8.22.2.6 m_PriBuildId . . . 45
8.22.2.7 m_PriImageId . . . 46
8.23 cdmaSSInfo Struct Reference . . . 46
8.23.1 Detailed Description . . . 46
8.23.2 Field Documentation . . . 46
8.23.2.1 ecio . . . 46
8.23.2.2 rssi . . . 46
8.24 connectionStatus Struct Reference . . . 46
8.24.1 Detailed Description . . . 46
8.24.2 Field Documentation . . . 46
8.24.2.1 MDMCallDuration . . . 46
8.24.2.2 MDMConnStatus . . . 46
8.25 crashInfoParams Struct Reference . . . 46
8.25.1 Detailed Description . . . 47
8.25.2 Field Documentation . . . 47
8.25.2.1 crashInfo . . . 47
8.25.2.2 crashStatus . . . 47
8.26 crashInformation Struct Reference . . . 47
8.26.1 Detailed Description . . . 47
8.26.2 Field Documentation . . . 48
8.26.2.1 crashData . . . 48
8.26.2.2 crashId . . . 48
8.26.2.3 crashString . . . 48
8.26.2.4 crashStrlen . . . 48
8.26.2.5 gcdumpString . . . 48
8.26.2.6 gcdumpStrlen . . . 48
8.26.2.7 numCrashes . . . 48
8.27 currNetworkInfo Struct Reference . . . 48
8.27.1 Detailed Description . . . 48
8.27.2 Field Documentation . . . 48
8.27.2.1 NetworkType . . . 48
8.27.2.2 RATMask . . . 48
8.27.2.3 SOMask . . . 48
8.28 dms_ActivationStatusTlv Struct Reference . . . 48
8.28.1 Detailed Description . . . 49
8.28.2 Field Documentation . . . 49
8.28.2.1 activationStatus . . . 49
8.28.2.2 TlvPresent . . . 49

8.29 dms_OperatingModeTlv Struct Reference . . . 49
8.29.1 Detailed Description . . . 49
8.29.2 Field Documentation . . . 50
8.29.2.1 operatingMode . . . 50
8.29.2.2 TlvPresent . . . 50
8.30 DMScustSettingInfo Struct Reference . . . 50
8.30.1 Detailed Description . . . 50
8.30.2 Field Documentation . . . 51
8.30.2.1 cust_attr . . . 51
8.30.2.2 cust_id . . . 51
8.30.2.3 cust_value . . . 51
8.30.2.4 id_length . . . 51
8.30.2.5 value_length . . . 51
8.31 DMScustSettingList Struct Reference . . . 51
8.31.1 Detailed Description . . . 51
8.31.2 Field Documentation . . . 51
8.31.2.1 custSetting . . . 51
8.31.2.2 list_type . . . 51
8.31.2.3 num_instances . . . 51
8.32 DMSgetCustomFeatureV2 Struct Reference . . . 51
8.32.1 Detailed Description . . . 52
8.32.2 Field Documentation . . . 52
8.32.2.1 pCustSettingInfo . . . 52
8.32.2.2 pCustSettingList . . . 52
8.32.2.3 pGetCustomInput . . . 52
8.33 DMSgetCustomInput Struct Reference . . . 52
8.33.1 Detailed Description . . . 52
8.33.2 Field Documentation . . . 52
8.33.2.1 cust_id . . . 52
8.33.2.2 list_type . . . 52
8.34 dunchannelRate Struct Reference . . . 52
8.34.1 Detailed Description . . . 53
8.34.2 Field Documentation . . . 53
8.34.2.1 CurrChanRxRate . . . 53
8.34.2.2 CurrChanTxRate . . . 53
8.34.2.3 MaxChanRxRate . . . 53
8.34.2.4 MaxChanTxRate . . . 53
8.35 eriDataparams Struct Reference . . . 53
8.35.1 Field Documentation . . . 53
8.35.1.1 eriData . . . 53

8.35.1.2 eriDataLen . . . 53
8.36 eTWSPLMNInfoTlv Struct Reference . . . 53
8.36.1 Detailed Description . . . 53
8.36.2 Field Documentation . . . 54
8.36.2.1 ETWSPLMNInfo . . . 54
8.36.2.2 TlvPresent . . . 54
8.37 FMSImageElement Struct Reference . . . 54
8.37.1 Detailed Description . . . 54
8.37.2 Field Documentation . . . 54
8.37.2.1 buildId . . . 54
8.37.2.2 buildIdLength . . . 54
8.37.2.3 imageId . . . 54
8.37.2.4 imageType . . . 54
8.38 FMSImageIdElement Struct Reference . . . 55
8.38.1 Detailed Description . . . 55
8.38.2 Field Documentation . . . 55
8.38.2.1 buildID . . . 55
8.38.2.2 buildIDLength . . . 55
8.38.2.3 failureCount . . . 55
8.38.2.4 imageID . . . 55
8.38.2.5 storageIndex . . . 55
8.39 FMSImageIDEntries Struct Reference . . . 55
8.39.1 Detailed Description . . . 56
8.39.2 Field Documentation . . . 56
8.39.2.1 executingImage . . . 56
8.39.2.2 imageIDElement . . . 56
8.39.2.3 imageIDSize . . . 56
8.39.2.4 imageType . . . 56
8.39.2.5 maxImages . . . 56
8.40 FMSImageList Struct Reference . . . 56
8.40.1 Detailed Description . . . 56
8.40.2 Field Documentation . . . 57
8.40.2.1 imageIDEntries . . . 57
8.40.2.2 listSize . . . 57
8.41 FMSPrefImageList Struct Reference . . . 57
8.41.1 Detailed Description . . . 57
8.41.2 Field Documentation . . . 57
8.41.2.1 listEntries . . . 57
8.41.2.2 listSize . . . 57
8.42 hdrSSInfo Struct Reference . . . 57

8.42.1 Detailed Description . . . 57
8.42.2 Field Documentation . . . 58
8.42.2.1 ecio . . . 58
8.42.2.2 io . . . 58
8.42.2.3 rssi . . . 58
8.42.2.4 sinr . . . 58
8.43 image_info_t Struct Reference . . . 58
8.43.1 Field Documentation . . . 58
8.43.1.1 buildID . . . 58
8.43.1.2 buildIDLen . . . 58
8.43.1.3 imageType . . . 58
8.43.1.4 uniqueID . . . 58
8.44 ipv6AddressInfo Struct Reference . . . 58
8.44.1 Detailed Description . . . 58
8.44.2 Field Documentation . . . 58
8.44.2.1 IPAddressV6 . . . 59
8.44.2.2 IPV6PrefixLen . . . 59
8.45 LibPackGPRSRequestedQoS Struct Reference . . . 59
8.45.1 Detailed Description . . . 59
8.45.2 Field Documentation . . . 59
8.45.2.1 delayClass . . . 59
8.45.2.2 meanThroughputClass . . . 59
8.45.2.3 peakThroughputClass . . . 59
8.45.2.4 precedenceClass . . . 59
8.45.2.5 reliabilityClass . . . 59
8.46 LibpackProfile3GPP Struct Reference . . . 59
8.46.1 Detailed Description . . . 60
8.46.2 Field Documentation . . . 64
8.46.2.1 pAddrAllocPref . . . 64
8.46.2.2 pAPNClass . . . 64
8.46.2.3 pAPNDisabledFlag . . . 64
8.46.2.4 pAPNName . . . 64
8.46.2.5 pAPNnameSize . . . 64
8.46.2.6 pAuthenticationPref . . . 64
8.46.2.7 pGPRSMinimumQoS . . . 64
8.46.2.8 pGPRSRequestedQos . . . 64
8.46.2.9 pImCnFlag . . . 64
8.46.2.10 pIPv4AddrPref . . . 64
8.46.2.11 pIPv6AddPref . . . 64
8.46.2.12 pPassword . . . 64

8.46.2.13 pPasswordSize . . . 64
8.46.2.14 pPcscfAddrUsingDhcp . . . 64
8.46.2.15 pPcscfAddrUsingPCO . . . 64
8.46.2.16 pPDNInactivTimeout . . . 64
8.46.2.17 pPdpAccessConFlag . . . 64
8.46.2.18 pPdpContext . . . 64
8.46.2.19 pPdpDataCompType . . . 65
8.46.2.20 pPdpHdrCompType . . . 65
8.46.2.21 pPDPtype . . . 65
8.46.2.22 pPriDNSIPv4AddPref . . . 65
8.46.2.23 pPriDNSIPv6addpref . . . 65
8.46.2.24 pPrimaryID . . . 65
8.46.2.25 pProfilename . . . 65
8.46.2.26 pProfilenameSize . . . 65
8.46.2.27 pQosClassID . . . 65
8.46.2.28 pSecDNSIPv4AddPref . . . 65
8.46.2.29 pSecDNSIPv6addpref . . . 65
8.46.2.30 pSecondaryFlag . . . 65
8.46.2.31 pTFTID1Params . . . 65
8.46.2.32 pTFTID2Params . . . 65
8.46.2.33 pUMTSMinQoS . . . 65
8.46.2.34 pUMTSMinQosSigInd . . . 65
8.46.2.35 pUMTSReqQoS . . . 65
8.46.2.36 pUMTSReqQoSSigInd . . . 65
8.46.2.37 pUsername . . . 65
8.46.2.38 pUsernameSize . . . 65
8.47 LibpackProfile3GPP2 Struct Reference . . . 65
8.47.1 Detailed Description . . . 66
8.47.2 Field Documentation . . . 69
8.47.2.1 pAllowLinger . . . 69
8.47.2.2 pAPNClass3GPP2 . . . 69
8.47.2.3 pAPNEnabled3GPP2 . . . 69
8.47.2.4 pApnString . . . 69
8.47.2.5 pApnStringSize . . . 70
8.47.2.6 pAppPriority . . . 70
8.47.2.7 pAppType . . . 70
8.47.2.8 pAuthPassword . . . 70
8.47.2.9 pAuthPasswordSize . . . 70
8.47.2.10 pAuthProtocol . . . 70
8.47.2.11 pAuthRetryCount . . . 70

8.47.2.12 pAuthTimeout . . . 70
8.47.2.13 pDataMode . . . 70
8.47.2.14 pDataRate . . . 70
8.47.2.15 pIpcpAckTimeout . . . 70
8.47.2.16 pIpcpCreqRetryCount . . . 70
8.47.2.17 pIsPcscfAddressNedded . . . 70
8.47.2.18 pLcpAckTimeout . . . 70
8.47.2.19 pLcpCreqRetryCount . . . 70
8.47.2.20 pNegoDnsSrvrPref . . . 70
8.47.2.21 pPDNInactivTimeout3GPP2 . . . 70
8.47.2.22 pPdnType . . . 70
8.47.2.23 pPppSessCloseTimer1x . . . 70
8.47.2.24 pPppSessCloseTimerDO . . . 70
8.47.2.25 pPrimaryV4DnsAddress . . . 70
8.47.2.26 pPriV6DnsAddress . . . 70
8.47.2.27 pRATType . . . 70
8.47.2.28 pSecondaryV4DnsAddress . . . 70
8.47.2.29 pSecV6DnsAddress . . . 70
8.47.2.30 pUserId . . . 70
8.47.2.31 pUserIdSize . . . 70
8.48 LibPackprofile_3GPP Struct Reference . . . 70
8.48.1 Detailed Description . . . 71
8.48.2 Field Documentation . . . 75
8.48.2.1 pAddrAllocPref . . . 75
8.48.2.2 pAPNClass . . . 75
8.48.2.3 pAPNDisabledFlag . . . 75
8.48.2.4 pAPNName . . . 75
8.48.2.5 pAPNnameSize . . . 75
8.48.2.6 pAuthenticationPref . . . 75
8.48.2.7 pGPRSMinimumQoS . . . 75
8.48.2.8 pGPRSRequestedQos . . . 75
8.48.2.9 pImCnFlag . . . 75
8.48.2.10 pIPv4AddrPref . . . 75
8.48.2.11 pIPv6AddPref . . . 75
8.48.2.12 pPassword . . . 75
8.48.2.13 pPasswordSize . . . 75
8.48.2.14 pPcscfAddrUsingDhcp . . . 76
8.48.2.15 pPcscfAddrUsingPCO . . . 76
8.48.2.16 pPDNInactivTimeout . . . 76
8.48.2.17 pPdpAccessConFlag . . . 76

8.48.2.18 pPdpContext . . . 76
8.48.2.19 pPdpDataCompType . . . 76
8.48.2.20 pPdpHdrCompType . . . 76
8.48.2.21 pPDPtype . . . 76
8.48.2.22 pPriDNSIPv4AddPref . . . 76
8.48.2.23 pPriDNSIPv6addpref . . . 76
8.48.2.24 pPrimaryID . . . 76
8.48.2.25 pProfilename . . . 76
8.48.2.26 pProfilenameSize . . . 76
8.48.2.27 pQosClassID . . . 76
8.48.2.28 pSecDNSIPv4AddPref . . . 76
8.48.2.29 pSecDNSIPv6addpref . . . 76
8.48.2.30 pSecondaryFlag . . . 76
8.48.2.31 pTFTID1Params . . . 76
8.48.2.32 pTFTID2Params . . . 76
8.48.2.33 pUMTSMinQoS . . . 76
8.48.2.34 pUMTSMinQosSigInd . . . 76
8.48.2.35 pUMTSReqQoS . . . 76
8.48.2.36 pUMTSReqQoSSigInd . . . 76
8.48.2.37 pUsername . . . 76
8.48.2.38 pUsernameSize . . . 76
8.49 LibPackprofile_3GPP2 Struct Reference . . . 76
8.49.1 Detailed Description . . . 77
8.49.2 Field Documentation . . . 80
8.49.2.1 pAllowLinger . . . 80
8.49.2.2 pAPNClass3GPP2 . . . 80
8.49.2.3 pAPNEnabled3GPP2 . . . 81
8.49.2.4 pApnString . . . 81
8.49.2.5 pApnStringSize . . . 81
8.49.2.6 pAppPriority . . . 81
8.49.2.7 pAppType . . . 81
8.49.2.8 pAuthPassword . . . 81
8.49.2.9 pAuthPassword_tSize . . . 81
8.49.2.10 pAuthProtocol . . . 81
8.49.2.11 pAuthRetryCount . . . 81
8.49.2.12 pAuthTimeout . . . 81
8.49.2.13 pDataMode . . . 81
8.49.2.14 pDataRate . . . 81
8.49.2.15 pIpcpAckTimeout . . . 81
8.49.2.16 pIpcpCreqRetryCount . . . 81

8.49.2.17 pIsPcscfAddressNedded . . . 81
8.49.2.18 pLcpAckTimeout . . . 81
8.49.2.19 pLcpCreqRetryCount . . . 81
8.49.2.20 pNegoDnsSrvrPref . . . 81
8.49.2.21 pPDNInactivTimeout3GPP2 . . . 81
8.49.2.22 pPdnType . . . 81
8.49.2.23 pPppSessCloseTimer1x . . . 81
8.49.2.24 pPppSessCloseTimerDO . . . 81
8.49.2.25 pPrimaryV4DnsAddress . . . 81
8.49.2.26 pPriV6DnsAddress . . . 81
8.49.2.27 pRATType . . . 81
8.49.2.28 pSecondaryV4DnsAddress . . . 81
8.49.2.29 pSecV6DnsAddress . . . 81
8.49.2.30 pUserId . . . 81
8.49.2.31 pUserIdSize . . . 82
8.50 LibPackQosClassID Struct Reference . . . 82
8.50.1 Detailed Description . . . 82
8.50.2 Field Documentation . . . 82
8.50.2.1 gDlBitRate . . . 82
8.50.2.2 gUlBitRate . . . 82
8.50.2.3 maxDlBitRate . . . 82
8.50.2.4 maxUlBitRate . . . 82
8.50.2.5 QCI . . . 82
8.51 LibPackTFTIDParams Struct Reference . . . 82
8.51.1 Detailed Description . . . 83
8.51.2 Field Documentation . . . 84
8.51.2.1 destPortRangeEnd . . . 84
8.51.2.2 destPortRangeStart . . . 84
8.51.2.3 eValid . . . 84
8.51.2.4 filterId . . . 84
8.51.2.5 flowLabel . . . 84
8.51.2.6 IPSECSPI . . . 84
8.51.2.7 ipVersion . . . 84
8.51.2.8 nextHeader . . . 84
8.51.2.9 pSourceIP . . . 84
8.51.2.10 sourceIPMask . . . 84
8.51.2.11 srcPortRangeEnd . . . 84
8.51.2.12 srcPortRangeStart . . . 84
8.51.2.13 tosMask . . . 84
8.52 LibPackUMTSQoS Struct Reference . . . 84

8.52.1 Detailed Description . . . 85
8.52.2 Field Documentation . . . 86
8.52.2.1 deliveryErrSDU . . . 86
8.52.2.2 grntDownlinkBitrate . . . 86
8.52.2.3 grntUplinkBitrate . . . 86
8.52.2.4 maxDownlinkBitrate . . . 86
8.52.2.5 maxSDUSize . . . 86
8.52.2.6 maxUplinkBitrate . . . 86
8.52.2.7 qosDeliveryOrder . . . 86
8.52.2.8 resBerRatio . . . 87
8.52.2.9 sduErrorRatio . . . 87
8.52.2.10 trafficClass . . . 87
8.52.2.11 trafficPriority . . . 87
8.52.2.12 transferDelay . . . 87
8.53 LibPackUMTSReqQoSSigInd Struct Reference . . . 87
8.53.1 Detailed Description . . . 87
8.53.2 Field Documentation . . . 87
8.53.2.1 SigInd . . . 87
8.53.2.2 UMTSReqQoS . . . 87
8.54 loc_BdsSV Struct Reference . . . 87
8.54.1 Detailed Description . . . 87
8.54.2 Field Documentation . . . 88
8.54.2.1 id . . . 88
8.54.2.2 mask . . . 88
8.55 loc_BdsSVInfo Struct Reference . . . 88
8.55.1 Detailed Description . . . 88
8.55.2 Field Documentation . . . 88
8.55.2.1 len . . . 88
8.55.2.2 pSV . . . 88
8.56 loc_CellDb Struct Reference . . . 88
8.56.1 Detailed Description . . . 89
8.56.2 Field Documentation . . . 89
8.56.2.1 mask . . . 89
8.57 loc_ClkInfo Struct Reference . . . 89
8.57.1 Detailed Description . . . 89
8.57.2 Field Documentation . . . 90
8.57.2.1 mask . . . 90
8.58 loc_GnssData Struct Reference . . . 90
8.58.1 Detailed Description . . . 91
8.58.2 Field Documentation . . . 91

8.58.2.1 mask . . . 92
8.59 loc_gpsTime Struct Reference . . . 92
8.59.1 Detailed Description . . . 92
8.59.2 Field Documentation . . . 92
8.59.2.1 gpsTimeOfWeekMs . . . 92
8.59.2.2 gpsWeek . . . 92
8.60 loc_LocApplicationInfo Struct Reference . . . 92
8.60.1 Detailed Description . . . 92
8.60.2 Field Documentation . . . 93
8.60.2.1 appNameLength . . . 93
8.60.2.2 appProviderLength . . . 93
8.60.2.3 appVersionLength . . . 93
8.60.2.4 appVersionValid . . . 93
8.60.2.5 pAppName . . . 93
8.60.2.6 pAppProvider . . . 93
8.60.2.7 pAppVersion . . . 93
8.61 loc_precisionDilution Struct Reference . . . 93
8.61.1 Detailed Description . . . 93
8.61.2 Field Documentation . . . 94
8.61.2.1 HDOP . . . 94
8.61.2.2 PDOP . . . 94
8.61.2.3 VDOP . . . 94
8.62 loc_satelliteInfo Struct Reference . . . 94
8.62.1 Detailed Description . . . 94
8.62.2 Field Documentation . . . 96
8.62.2.1 azimuth . . . 96
8.62.2.2 elevation . . . 96
8.62.2.3 gnssSvId . . . 96
8.62.2.4 healthStatus . . . 96
8.62.2.5 snr . . . 96
8.62.2.6 svInfoMask . . . 96
8.62.2.7 svListLen . . . 96
8.62.2.8 svStatus . . . 96
8.62.2.9 system . . . 96
8.62.2.10 validMask . . . 96
8.63 loc_sensorDataUsage Struct Reference . . . 96
8.63.1 Detailed Description . . . 96
8.63.2 Field Documentation . . . 97
8.63.2.1 aidingIndicatorMask . . . 97
8.63.2.2 usageMask . . . 97

8.64 loc_SV Struct Reference . . . 97
8.64.1 Detailed Description . . . 97
8.64.2 Field Documentation . . . 98
8.64.2.1 id . . . 98
8.64.2.2 mask . . . 98
8.64.2.3 system . . . 98
8.65 loc_SVInfo Struct Reference . . . 98
8.65.1 Detailed Description . . . 98
8.65.2 Field Documentation . . . 98
8.65.2.1 len . . . 99
8.65.2.2 pSV . . . 99
8.66 loc_svUsedforFix Struct Reference . . . 99
8.66.1 Detailed Description . . . 99
8.66.2 Field Documentation . . . 99
8.66.2.1 gnssSvUsedList . . . 99
8.66.2.2 gnssSvUsedList_len . . . 99
8.67 lteSSInfo Struct Reference . . . 99
8.67.1 Detailed Description . . . 99
8.67.2 Field Documentation . . . 100
8.67.2.1 rsrp . . . 100
8.67.2.2 rsrq . . . 100
8.67.2.3 rssi . . . 100
8.67.2.4 snr . . . 100
8.68 messageModeTlv Struct Reference . . . 100
8.68.1 Detailed Description . . . 100
8.68.2 Field Documentation . . . 100
8.68.2.1 MessageModeInfo . . . 100
8.68.2.2 TlvPresent . . . 100
8.69 nas_acqOrderPref Struct Reference . . . 100
8.69.1 Detailed Description . . . 100
8.69.2 Field Documentation . . . 101
8.69.2.1 acqOrdeLen . . . 101
8.69.2.2 pAcqOrder . . . 101
8.70 nas_ActPilotPNElement Struct Reference . . . 101
8.70.1 Detailed Description . . . 101
8.70.2 Field Documentation . . . 101
8.70.2.1 ActSetPilotPN . . . 101
8.70.2.2 ActSetPilotPNStrength . . . 101
8.71 nas_AddCDMASysInfo Struct Reference . . . 101
8.71.1 Detailed Description . . . 101

8.71.2 Field Documentation . . . 102
8.71.2.1 geoSysIdx . . . 102
8.71.2.2 regPrd . . . 102
8.72 nas_AddSysInfo Struct Reference . . . 102
8.72.1 Detailed Description . . . 102
8.72.2 Field Documentation . . . 102
8.72.2.1 cellBroadcastCap . . . 102
8.72.2.2 geoSysIdx . . . 102
8.73 nas_CallBarringSysInfo Struct Reference . . . 102
8.73.1 Detailed Description . . . 103
8.73.2 Field Documentation . . . 103
8.73.2.1 csBarStatus . . . 103
8.73.2.2 psBarStatus . . . 103
8.74 nas_callBarStatus Struct Reference . . . 103
8.74.1 Detailed Description . . . 103
8.74.2 Field Documentation . . . 104
8.74.2.1 csBarStatus . . . 104
8.74.2.2 psBarStatus . . . 104
8.75 nas_CDMAChannel Struct Reference . . . 104
8.75.1 Detailed Description . . . 104
8.75.2 Field Documentation . . . 105
8.75.2.1 priChA . . . 105
8.75.2.2 priChB . . . 105
8.75.2.3 secChA . . . 105
8.75.2.4 secChB . . . 105
8.76 nas_CDMAECIOThresh Struct Reference . . . 105
8.76.1 Detailed Description . . . 105
8.76.2 Field Documentation . . . 105
8.76.2.1 CDMAECIOThreshListLen . . . 105
8.76.2.2 pCDMAECIOThreshList . . . 105
8.77 nas_CDMAInfo Struct Reference . . . 105
8.77.1 Detailed Description . . . 106
8.77.2 Field Documentation . . . 106
8.77.2.1 baseId . . . 106
8.77.2.2 baseLat . . . 106
8.77.2.3 baseLong . . . 106
8.77.2.4 nid . . . 106
8.77.2.5 refpn . . . 106
8.77.2.6 sid . . . 106
8.78 nas_CDMARSSIThresh Struct Reference . . . 106

8.78.1 Detailed Description . . . 107
8.78.2 Field Documentation . . . 107
8.78.2.1 CDMARSSIThreshListLen . . . 107
8.78.2.2 pCDMARSSIThreshList . . . 107
8.79 nas_CDMASysInfo Struct Reference . . . 107
8.79.1 Detailed Description . . . 107
8.79.2 Field Documentation . . . 110
8.79.2.1 baseId . . . 110
8.79.2.2 baseLat . . . 110
8.79.2.3 baseLong . . . 110
8.79.2.4 bsInfoValid . . . 110
8.79.2.5 bsPRev . . . 110
8.79.2.6 bsPRevValid . . . 110
8.79.2.7 ccsSupported . . . 110
8.79.2.8 ccsSupportedValid . . . 110
8.79.2.9 cdmaSysIdValid . . . 110
8.79.2.10 isSysPrlMatch . . . 110
8.79.2.11 isSysPrlMatchValid . . . 110
8.79.2.12 MCC . . . 110
8.79.2.13 MNC . . . 110
8.79.2.14 networkID . . . 110
8.79.2.15 networkIdValid . . . 110
8.79.2.16 packetZone . . . 110
8.79.2.17 packetZoneValid . . . 110
8.79.2.18 pRevInUse . . . 111
8.79.2.19 pRevInUseValid . . . 111
8.79.2.20 sysInfoCDMA . . . 111
8.79.2.21 systemID . . . 111
8.80 nas_CDMASysInfoExt Struct Reference . . . 111
8.80.1 Detailed Description . . . 111
8.80.2 Field Documentation . . . 111
8.80.2.1 imsi_11_12 . . . 111
8.80.2.2 MCC . . . 111
8.81 nas_cellParams Struct Reference . . . 111
8.81.1 Detailed Description . . . 111
8.81.2 Field Documentation . . . 112
8.81.2.1 pci . . . 112
8.81.2.2 rsrp . . . 112
8.81.2.3 rsrq . . . 112
8.81.2.4 rssi . . . 112

8.81.2.5 srxlev . . . 112
8.82 nas_CommInfo Struct Reference . . . 112
8.82.1 Detailed Description . . . 112
8.82.2 Field Documentation . . . 114
8.82.2.1 imsRegState . . . 114
8.82.2.2 modemMode . . . 114
8.82.2.3 psState . . . 114
8.82.2.4 systemMode . . . 114
8.82.2.5 temperature . . . 114
8.83 nas_CSGID Struct Reference . . . 114
8.83.1 Detailed Description . . . 114
8.83.2 Field Documentation . . . 114
8.83.2.1 id . . . 115
8.83.2.2 mcc . . . 115
8.83.2.3 mnc . . . 115
8.83.2.4 mncPcsDigits . . . 115
8.83.2.5 rat . . . 115
8.84 nas_currentPLMN Struct Reference . . . 115
8.84.1 Detailed Description . . . 115
8.84.2 Field Documentation . . . 115
8.84.2.1 MCC . . . 115
8.84.2.2 MNC . . . 115
8.84.2.3 netDescr . . . 116
8.84.2.4 netDescrLength . . . 116
8.85 nas_dataSrvCapabilities Struct Reference . . . 116
8.85.1 Detailed Description . . . 116
8.85.2 Field Documentation . . . 116
8.85.2.1 dataCapabilities . . . 116
8.85.2.2 dataCapabilitiesLen . . . 116
8.86 nas_DataStatusDetail Struct Reference . . . 116
8.86.1 Detailed Description . . . 117
8.86.2 Field Documentation . . . 118
8.86.2.1 IPAddress . . . 118
8.86.2.2 LastErrCode . . . 118
8.87 nas_detailSvcInfo Struct Reference . . . 119
8.87.1 Detailed Description . . . 119
8.87.2 Field Documentation . . . 120
8.87.2.1 hdrHybrid . . . 120
8.87.2.2 hdrSrvStatus . . . 120
8.87.2.3 isSysForbidden . . . 120

8.87.2.4 srvCapability . . . 120
8.87.2.5 srvStatus . . . 120
8.88 nas_DeviceConfigDetail Struct Reference . . . 120
8.88.1 Detailed Description . . . 120
8.88.2 Field Documentation . . . 121
8.88.2.1 Chipset . . . 121
8.88.2.2 HWVersion . . . 121
8.88.2.3 QLIC . . . 121
8.88.2.4 Technology . . . 121
8.89 nas_dirNum Struct Reference . . . 121
8.89.1 Detailed Description . . . 121
8.89.2 Field Documentation . . . 121
8.89.2.1 dirNum . . . 121
8.89.2.2 dirNumLen . . . 121
8.90 nas_DRCParams Struct Reference . . . 121
8.90.1 Detailed Description . . . 122
8.90.2 Field Documentation . . . 122
8.90.2.1 DRCCover . . . 122
8.90.2.2 DRCValue . . . 122
8.91 nas_ecioListElement Struct Reference . . . 122
8.91.1 Detailed Description . . . 122
8.91.2 Field Documentation . . . 122
8.91.2.1 ecio . . . 122
8.91.2.2 radioIf . . . 123
8.92 nas_ECIOThresh Struct Reference . . . 123
8.92.1 Detailed Description . . . 123
8.92.2 Field Documentation . . . 123
8.92.2.1 ECIOThresListLen . . . 123
8.92.2.2 pECIOThresList . . . 123
8.93 nas_errorRateListElement Struct Reference . . . 123
8.93.1 Detailed Description . . . 123
8.93.2 Field Documentation . . . 124
8.93.2.1 errorRate . . . 124
8.93.2.2 radioIf . . . 124
8.94 nas_GERANInfo Struct Reference . . . 124
8.94.1 Detailed Description . . . 125
8.94.2 Field Documentation . . . 126
8.94.2.1 arfcn . . . 126
8.94.2.2 bsic . . . 126
8.94.2.3 cellID . . . 126

8.94.2.4 insNmrCellInfo . . . 126
8.94.2.5 lac . . . 126
8.94.2.6 nmrInst . . . 126
8.94.2.7 plmn . . . 126
8.94.2.8 rxLev . . . 126
8.94.2.9 timingAdvance . . . 126
8.95 nas_geranInstInfo Struct Reference . . . 126
8.95.1 Detailed Description . . . 126
8.95.2 Field Documentation . . . 126
8.95.2.1 geranArfcn . . . 126
8.95.2.2 geranBsicBcc . . . 127
8.95.2.3 geranBsicNcc . . . 127
8.95.2.4 geranRssi . . . 127
8.96 nas_gsmCellInfo Struct Reference . . . 127
8.96.1 Detailed Description . . . 127
8.96.2 Field Documentation . . . 127
8.96.2.1 arfcn . . . 128
8.96.2.2 band1900 . . . 128
8.96.2.3 bsicId . . . 128
8.96.2.4 cellIdValid . . . 128
8.96.2.5 rssi . . . 128
8.96.2.6 srxlev . . . 128
8.97 nas_GSMRSSIThresh Struct Reference . . . 128
8.97.1 Detailed Description . . . 128
8.97.2 Field Documentation . . . 128
8.97.2.1 GSMRSSIThreshListLen . . . 128
8.97.2.2 pGSMRSSIThreshList . . . 128
8.98 nas_GSMSrvStatusInfo Struct Reference . . . 128
8.98.1 Detailed Description . . . 128
8.98.2 Field Documentation . . . 129
8.98.2.1 isPrefDataPath . . . 129
8.98.2.2 srvStatus . . . 129
8.98.2.3 trueSrvStatus . . . 129
8.99 nas_GSMSysInfo Struct Reference . . . 129
8.99.1 Detailed Description . . . 130
8.99.2 Field Documentation . . . 131
8.99.2.1 cellId . . . 131
8.99.2.2 cellIdValid . . . 131
8.99.2.3 dtmSupp . . . 131
8.99.2.4 dtmSuppValid . . . 132

8.99.2.5 egprsSupp . . . 132
8.99.2.6 egprsSuppValid . . . 132
8.99.2.7 lac . . . 132
8.99.2.8 lacValid . . . 132
8.99.2.9 MCC . . . 132
8.99.2.10 MNC . . . 132
8.99.2.11 networkIdValid . . . 132
8.99.2.12 regRejectInfoValid . . . 132
8.99.2.13 rejCause . . . 132
8.99.2.14 rejectSrvDomain . . . 132
8.99.2.15 sysInfoGSM . . . 132
8.100 nas_HDRECIOThresh Struct Reference . . . 132
8.100.1 Detailed Description . . . 132
8.100.2 Field Documentation . . . 132
8.100.2.1 HDRECIOThreshListLen . . . 132
8.100.2.2 pHDRECIOThreshList . . . 132
8.101 nas_HDRIOThresh Struct Reference . . . 132
8.101.1 Detailed Description . . . 133
8.101.2 Field Documentation . . . 133
8.101.2.1 HDRIOThreshListLen . . . 133
8.101.2.2 pHDRIOThreshList . . . 133
8.102 nas_HDRRSSIThresh Struct Reference . . . 133
8.102.1 Detailed Description . . . 133
8.102.2 Field Documentation . . . 133
8.102.2.1 HDRRSSIThreshListLen . . . 133
8.102.2.2 pHDRRSSIThreshList . . . 134
8.103 nas_HDRSINRThresh Struct Reference . . . 134
8.103.1 Detailed Description . . . 134
8.103.2 Field Documentation . . . 134
8.103.2.1 HDRSINRThresListLen . . . 134
8.103.2.2 pHDRSINRThresList . . . 134
8.104 nas_HDRSINRThreshold Struct Reference . . . 134
8.104.1 Detailed Description . . . 134
8.104.2 Field Documentation . . . 135
8.104.2.1 HDRSINRThreshListLen . . . 135
8.104.2.2 pHDRSINRThreshList . . . 135
8.105 nas_HDRSysInfo Struct Reference . . . 135
8.105.1 Detailed Description . . . 135
8.105.2 Field Documentation . . . 137
8.105.2.1 hdrActiveProt . . . 137

8.105.2.2 hdrActiveProtValid . . . 137
8.105.2.3 hdrPersonality . . . 137
8.105.2.4 hdrPersonalityValid . . . 137
8.105.2.5 is856SysId . . . 137
8.105.2.6 is856SysIdValid . . . 137
8.105.2.7 isSysPrlMatch . . . 137
8.105.2.8 isSysPrlMatchValid . . . 137
8.105.2.9 sysInfoHDR . . . 137
8.106 nas_homeSIDNID Struct Reference . . . 137
8.106.1 Detailed Description . . . 137
8.106.2 Field Documentation . . . 137
8.106.2.1 numInstances . . . 137
8.106.2.2 SidNid . . . 137
8.107 nas_infoInterFreq Struct Reference . . . 138
8.107.1 Detailed Description . . . 138
8.107.2 Field Documentation . . . 138
8.107.2.1 cell_resel_priority . . . 138
8.107.2.2 cellInterFreqParams . . . 138
8.107.2.3 cells_len . . . 138
8.107.2.4 earfcn . . . 139
8.107.2.5 threshXHigh . . . 139
8.107.2.6 threshXLow . . . 139
8.108 nas_IOThresh Struct Reference . . . 139
8.108.1 Detailed Description . . . 139
8.108.2 Field Documentation . . . 139
8.108.2.1 IOThresListLen . . . 139
8.108.2.2 pIOThresList . . . 139
8.109 nas_lteEARFCN Struct Reference . . . 139
8.109.1 Detailed Description . . . 139
8.109.2 Field Documentation . . . 140
8.109.2.1 earfcn0 . . . 140
8.109.2.2 earfcn1 . . . 140
8.109.2.3 status . . . 140
8.110 nas_lteGsmCellInfo Struct Reference . . . 140
8.110.1 Detailed Description . . . 140
8.110.2 Field Documentation . . . 141
8.110.2.1 cellReselPriority . . . 141
8.110.2.2 cells_len . . . 141
8.110.2.3 GsmCellInfo . . . 141
8.110.2.4 nccPermitted . . . 141

8.110.2.5 threshGsmHigh . . . 141
8.110.2.6 threshGsmLow . . . 141
8.111 nas_LTEInfo Struct Reference . . . 141
8.111.1 Detailed Description . . . 141
8.111.2 Field Documentation . . . 143
8.111.2.1 band . . . 143
8.111.2.2 bandwidth . . . 143
8.111.2.3 emmConnState . . . 143
8.111.2.4 emmState . . . 143
8.111.2.5 emmSubState . . . 143
8.111.2.6 RXChan . . . 143
8.111.2.7 TXChan . . . 143
8.112 nas_LTEInfoInterfreq Struct Reference . . . 144
8.112.1 Detailed Description . . . 144
8.112.2 Field Documentation . . . 144
8.112.2.1 freqsLen . . . 144
8.112.2.2 InfoInterfreq . . . 144
8.112.2.3 ueInIdle . . . 144
8.113 nas_LTEInfoIntrafreq Struct Reference . . . 144
8.113.1 Detailed Description . . . 145
8.113.2 Field Documentation . . . 146
8.113.2.1 CellParams . . . 146
8.113.2.2 cellReselPriority . . . 146
8.113.2.3 cellsLen . . . 146
8.113.2.4 earfcn . . . 146
8.113.2.5 globalCellId . . . 146
8.113.2.6 plmn . . . 146
8.113.2.7 servingCellId . . . 146
8.113.2.8 sIntraSearch . . . 146
8.113.2.9 sNonIntraSearch . . . 146
8.113.2.10 tac . . . 146
8.113.2.11 threshServingLow . . . 146
8.113.2.12 ueInIdle . . . 146
8.114 nas_LTEInfoNeighboringGSM Struct Reference . . . 146
8.114.1 Detailed Description . . . 147
8.114.2 Field Documentation . . . 147
8.114.2.1 freqsLen . . . 147
8.114.2.2 LteGsmCellInfo . . . 147
8.114.2.3 ueInIdle . . . 147
8.115 nas_LTEInfoNeighboringWCDMA Struct Reference . . . 147

8.115.1 Detailed Description . . . 147
8.115.2 Field Documentation . . . 148
8.115.2.1 freqsLen . . . 148
8.115.2.2 LTEWCDMACellInfo . . . 148
8.115.2.3 ueInIdle . . . 148
8.116 nas_ltePCI Struct Reference . . . 148
8.116.1 Detailed Description . . . 148
8.116.2 Field Documentation . . . 148
8.116.2.1 earfcn . . . 148
8.116.2.2 pci . . . 148
8.116.2.3 status . . . 148
8.117 nas_lteRsrpinformation Struct Reference . . . 148
8.117.1 Detailed Description . . . 148
8.117.2 Field Documentation . . . 149
8.117.2.1 rsrplevel . . . 149
8.118 nas_LTERSRPThresh Struct Reference . . . 149
8.118.1 Detailed Description . . . 149
8.118.2 Field Documentation . . . 149
8.118.2.1 LTERSRPThreshListLen . . . 149
8.118.2.2 pLTERSRPThreshList . . . 149
8.119 nas_LTERSRQThresh Struct Reference . . . 149
8.119.1 Detailed Description . . . 149
8.119.2 Field Documentation . . . 150
8.119.2.1 LTERSRQThreshListLen . . . 150
8.119.2.2 pLTERSRQThreshList . . . 150
8.120 nas_LTERSSIThresh Struct Reference . . . 150
8.120.1 Detailed Description . . . 150
8.120.2 Field Documentation . . . 150
8.120.2.1 LTERSSIThreshListLen . . . 150
8.120.2.2 pLTERSSIThreshList . . . 150
8.121 nas_LTESigRptCfg Struct Reference . . . 150
8.121.1 Detailed Description . . . 151
8.121.2 Field Documentation . . . 151
8.121.2.1 avgPeriod . . . 151
8.121.2.2 rptRate . . . 151
8.122 nas_LTESigRptConfig Struct Reference . . . 151
8.122.1 Detailed Description . . . 151
8.122.2 Field Documentation . . . 152
8.122.2.1 avgPeriod . . . 152
8.122.2.2 rptRate . . . 152

8.123 nas_lteSnrinformation Struct Reference . . . 152
8.123.1 Detailed Description . . . 152
8.123.2 Field Documentation . . . 152
8.123.2.1 snrlevel . . . 153
8.124 nas_LTESNRThresh Struct Reference . . . 153
8.124.1 Detailed Description . . . 153
8.124.2 Field Documentation . . . 153
8.124.2.1 LTESNRThresListLen . . . 153
8.124.2.2 pLTESNRThresList . . . 153
8.125 nas_LTESNRThreshold Struct Reference . . . 153
8.125.1 Detailed Description . . . 153
8.125.2 Field Documentation . . . 154
8.125.2.1 LTESNRThreshListLen . . . 154
8.125.2.2 pLTESNRThreshList . . . 154
8.126 nas_LTESysInfo Struct Reference . . . 154
8.126.1 Detailed Description . . . 154
8.126.2 Field Documentation . . . 156
8.126.2.1 cellId . . . 156
8.126.2.2 cellIdValid . . . 156
8.126.2.3 lac . . . 156
8.126.2.4 lacValid . . . 156
8.126.2.5 MCC . . . 156
8.126.2.6 MNC . . . 156
8.126.2.7 networkIdValid . . . 156
8.126.2.8 regRejectInfoValid . . . 156
8.126.2.9 rejCause . . . 156
8.126.2.10 rejectSrvDomain . . . 156
8.126.2.11 sysInfoLTE . . . 156
8.126.2.12 tac . . . 156
8.126.2.13 tacValid . . . 156
8.127 nas_lteWcdmaCellInfo Struct Reference . . . 156
8.127.1 Detailed Description . . . 157
8.127.2 Field Documentation . . . 157
8.127.2.1 cellReselPriority . . . 157
8.127.2.2 cellsLen . . . 157
8.127.2.3 threshXhigh . . . 157
8.127.2.4 threshXlow . . . 157
8.127.2.5 uarfcn . . . 157
8.127.2.6 WCDMACellInfo . . . 157
8.128 nas_minBasedIMSI Struct Reference . . . 157

8.128.1 Detailed Description . . . 158
8.128.2 Field Documentation . . . 158
8.128.2.1 imsiM1112 . . . 158
8.128.2.2 imsiMS1 . . . 158
8.128.2.3 imsiMS2 . . . 158
8.128.2.4 mccM . . . 158
8.129 nas_MNRInfo Struct Reference . . . 158
8.129.1 Detailed Description . . . 158
8.129.2 Field Documentation . . . 159
8.129.2.1 mcc . . . 159
8.129.2.2 mnc . . . 159
8.129.2.3 rat . . . 159
8.130 nas_namName Struct Reference . . . 159
8.130.1 Detailed Description . . . 159
8.130.2 Field Documentation . . . 159
8.130.2.1 namName . . . 159
8.130.2.2 namNameLen . . . 159
8.131 nas_netSelectionPref Struct Reference . . . 159
8.131.1 Detailed Description . . . 160
8.131.2 Field Documentation . . . 160
8.131.2.1 mcc . . . 160
8.131.2.2 mnc . . . 160
8.131.2.3 netReg . . . 160
8.132 nas_NetworkStat1x Struct Reference . . . 160
8.132.1 Detailed Description . . . 160
8.132.2 Field Documentation . . . 162
8.132.2.1 ActSetCnt . . . 162
8.132.2.2 NeighborSetCnt . . . 162
8.132.2.3 pActPilotPNElements . . . 162
8.132.2.4 pNeighborSetPilotPN . . . 162
8.132.2.5 RX_EC_IO . . . 162
8.132.2.6 RX_PWR . . . 162
8.132.2.7 SO . . . 162
8.132.2.8 State . . . 162
8.132.2.9 TX_PWR . . . 162
8.133 nas_NetworkStatEVDO Struct Reference . . . 162
8.133.1 Detailed Description . . . 162
8.133.2 Field Documentation . . . 163
8.133.2.1 MACIndex . . . 164
8.133.2.2 PER . . . 164

8.133.2.3 PilotEnergy . . . 164
8.133.2.4 pSectorID . . . 164
8.133.2.5 RX_PWR . . . 164
8.133.2.6 SectorIDLen . . . 164
8.133.2.7 SNR . . . 164
8.133.2.8 State . . . 164
8.134 nas_nmrCellInfo Struct Reference . . . 164
8.134.1 Detailed Description . . . 164
8.134.2 Field Documentation . . . 165
8.134.2.1 nmrArfcn . . . 165
8.134.2.2 nmrBsic . . . 165
8.134.2.3 nmrCellID . . . 165
8.134.2.4 nmrLac . . . 165
8.134.2.5 nmrPlmn . . . 165
8.134.2.6 nmrRxLev . . . 165
8.135 nas_operatorNameString Struct Reference . . . 165
8.135.1 Detailed Description . . . 165
8.135.2 Field Documentation . . . 165
8.135.2.1 PLMNName . . . 165
8.136 nas_OperatorPLMNData Struct Reference . . . 165
8.136.1 Detailed Description . . . 166
8.136.2 Field Documentation . . . 166
8.136.2.1 lac1 . . . 166
8.136.2.2 lac2 . . . 166
8.136.2.3 mcc . . . 166
8.136.2.4 mnc . . . 166
8.136.2.5 PLMNRecID . . . 166
8.137 nas_operatorPLMNList Struct Reference . . . 166
8.137.1 Detailed Description . . . 166
8.137.2 Field Documentation . . . 167
8.137.2.1 numInstance . . . 167
8.137.2.2 PLMNData . . . 167
8.138 nas_PhyCaAggPcellInfo Struct Reference . . . 167
8.138.1 Detailed Description . . . 167
8.138.2 Field Documentation . . . 167
8.138.2.1 dl_bw_value . . . 167
8.138.2.2 freq . . . 167
8.138.2.3 iLTEbandValue . . . 168
8.138.2.4 pci . . . 168
8.138.2.5 TlvPresent . . . 168

8.139 nas_PhyCaAggScellDlBw Struct Reference . . . 168
8.139.1 Detailed Description . . . 168
8.139.2 Field Documentation . . . 168
8.139.2.1 dl_bw_value . . . 168
8.139.2.2 TlvPresent . . . 168
8.140 nas_PhyCaAggScellIndex Struct Reference . . . 168
8.140.1 Detailed Description . . . 168
8.140.2 Field Documentation . . . 169
8.140.2.1 scell_idx . . . 169
8.140.2.2 TlvPresent . . . 169
8.141 nas_PhyCaAggScellIndType Struct Reference . . . 169
8.141.1 Detailed Description . . . 169
8.141.2 Field Documentation . . . 169
8.141.2.1 freq . . . 169
8.141.2.2 pci . . . 169
8.141.2.3 scell_state . . . 169
8.141.2.4 TlvPresent . . . 169
8.142 nas_PhyCaAggScellInfo Struct Reference . . . 169
8.142.1 Detailed Description . . . 170
8.142.2 Field Documentation . . . 172
8.142.2.1 dl_bw_value . . . 172
8.142.2.2 freq . . . 172
8.142.2.3 iLTEbandValue . . . 172
8.142.2.4 pci . . . 172
8.142.2.5 scell_state . . . 172
8.142.2.6 TlvPresent . . . 172
8.143 nas_PilotSetData Struct Reference . . . 172
8.143.1 Detailed Description . . . 172
8.143.2 Field Documentation . . . 172
8.143.2.1 NumPilots . . . 172
8.143.2.2 pPilotSetInfo . . . 172
8.144 nas_PilotSetParams Struct Reference . . . 172
8.144.1 Detailed Description . . . 173
8.144.2 Field Documentation . . . 173
8.144.2.1 PilotPN . . . 173
8.144.2.2 PilotStrength . . . 173
8.144.2.3 PilotType . . . 173
8.145 nas_PLMNNetworkName Struct Reference . . . 173
8.145.1 Detailed Description . . . 173
8.145.2 Field Documentation . . . 173

8.145.2.1 numInstance . . . 173
8.145.2.2 PLMNNetName . . . 173
8.146 nas_PLMNNetworkNameData Struct Reference . . . 174
8.146.1 Detailed Description . . . 174
8.146.2 Field Documentation . . . 175
8.146.2.1 codingScheme . . . 175
8.146.2.2 countryInitials . . . 175
8.146.2.3 longName . . . 175
8.146.2.4 longNameLen . . . 175
8.146.2.5 longNameSpareBits . . . 175
8.146.2.6 shortName . . . 175
8.146.2.7 shortNameLen . . . 175
8.146.2.8 shortNameSpareBits . . . 175
8.147 nas_protocolSubtypeElement Struct Reference . . . 175
8.147.1 Detailed Description . . . 175
8.147.2 Field Documentation . . . 176
8.147.2.1 AccessMac . . . 176
8.147.2.2 AuthProt . . . 176
8.147.2.3 ControlMac . . . 176
8.147.2.4 EncryptProt . . . 176
8.147.2.5 ForwardMac . . . 176
8.147.2.6 IdleState . . . 176
8.147.2.7 KeyExchange . . . 176
8.147.2.8 MultDisc . . . 177
8.147.2.9 PhysicalLayer . . . 177
8.147.2.10 ReverseMac . . . 177
8.147.2.11 SecProt . . . 177
8.147.2.12 VirtStream . . . 177
8.148 nas_qaQmi3Gpp2TimeZone Struct Reference . . . 177
8.148.1 Detailed Description . . . 177
8.148.2 Field Documentation . . . 177
8.148.2.1 daylightSavings . . . 177
8.148.2.2 leapSeconds . . . 177
8.148.2.3 localTimeOffset . . . 177
8.149 nas_QmiNas3GppNetworkInfo Struct Reference . . . 177
8.149.1 Detailed Description . . . 178
8.149.2 Field Documentation . . . 178
8.149.2.1 Desription . . . 178
8.149.2.2 Forbidden . . . 178
8.149.2.3 InUse . . . 178

8.149.2.4 MCC . . . 178
8.149.2.5 MNC . . . 178
8.149.2.6 Preferred . . . 178
8.149.2.7 Roaming . . . 178
8.150 nas_QmiNas3GppNetworkRAT Struct Reference . . . 178
8.150.1 Detailed Description . . . 178
8.150.2 Field Documentation . . . 179
8.150.2.1 MCC . . . 179
8.150.2.2 MNC . . . 179
8.150.2.3 RAT . . . 179
8.151 nas_QmisNasPcsDigit Struct Reference . . . 179
8.151.1 Detailed Description . . . 179
8.151.2 Field Documentation . . . 179
8.151.2.1 includes_pcs_digit . . . 179
8.151.2.2 MCC . . . 179
8.151.2.3 MNC . . . 179
8.152 nas_RejectReasonTlv Struct Reference . . . 180
8.152.1 Detailed Description . . . 180
8.152.2 Field Documentation . . . 180
8.152.2.1 rejectCause . . . 180
8.152.2.2 serviceDomain . . . 180
8.152.2.3 TlvPresent . . . 180
8.153 nas_RFInfoTlv Struct Reference . . . 180
8.153.1 Detailed Description . . . 180
8.153.2 Field Documentation . . . 180
8.153.2.1 activeBandClass . . . 180
8.153.2.2 activeChannel . . . 180
8.153.2.3 radioInterface . . . 180
8.153.2.4 radioInterfaceSize . . . 180
8.153.2.5 TlvPresent . . . 181
8.154 nas_roamIndList Struct Reference . . . 181
8.154.1 Detailed Description . . . 181
8.154.2 Field Documentation . . . 181
8.154.2.1 numInstances . . . 181
8.154.2.2 radioInterface . . . 181
8.154.2.3 roamIndicator . . . 181
8.155 nas_RSRPThresh Struct Reference . . . 181
8.155.1 Detailed Description . . . 182
8.155.2 Field Documentation . . . 182
8.155.2.1 pRSRPThresList . . . 182

8.155.2.2 RSRPThresListLen . . . 182
8.156 nas_rsrqInformation Struct Reference . . . 182
8.156.1 Detailed Description . . . 182
8.156.2 Field Documentation . . . 182
8.156.2.1 radioIf . . . 183
8.156.2.2 rsrq . . . 183
8.157 nas_RSRQThresh Struct Reference . . . 183
8.157.1 Detailed Description . . . 183
8.157.2 Field Documentation . . . 183
8.157.2.1 pRSRQThresList . . . 183
8.157.2.2 RSRQThresListLen . . . 183
8.158 nas_RSSIThresh Struct Reference . . . 183
8.158.1 Detailed Description . . . 183
8.158.2 Field Documentation . . . 184
8.158.2.1 pRSSIThresList . . . 184
8.158.2.2 RSSIThresListLen . . . 184
8.159 nas_rxInfo Struct Reference . . . 184
8.159.1 Detailed Description . . . 184
8.159.2 Field Documentation . . . 185
8.159.2.1 ecio . . . 185
8.159.2.2 isRadioTuned . . . 185
8.159.2.3 phase . . . 185
8.159.2.4 rscp . . . 185
8.159.2.5 rsrp . . . 185
8.159.2.6 rxPower . . . 185
8.160 nas_RxSigInfo Struct Reference . . . 185
8.160.1 Detailed Description . . . 185
8.160.2 Field Documentation . . . 186
8.160.2.1 isRadioTuned . . . 186
8.160.2.2 rsrp . . . 186
8.160.2.3 rxChainIndex . . . 186
8.160.2.4 rxPower . . . 186
8.161 nas_rxSignalStrengthListElement Struct Reference . . . 186
8.161.1 Detailed Description . . . 186
8.161.2 Field Documentation . . . 187
8.161.2.1 radioIf . . . 187
8.161.2.2 rxSignalStrength . . . 187
8.162 nas_SccRxInfo Struct Reference . . . 187
8.162.1 Detailed Description . . . 187
8.162.2 Field Documentation . . . 187

8.162.2.1 numInstances . . . 187
8.162.2.2 rsrq . . . 188
8.162.2.3 sigInfo . . . 188
8.162.2.4 snr . . . 188
8.162.2.5 TlvPresent . . . 188
8.163 nas_serviceProviderName Struct Reference . . . 188
8.163.1 Detailed Description . . . 188
8.163.2 Field Documentation . . . 188
8.163.2.1 displayCondition . . . 188
8.163.2.2 spn . . . 188
8.163.2.3 spnLength . . . 188
8.164 nas_servSystem Struct Reference . . . 188
8.164.1 Detailed Description . . . 188
8.164.2 Field Documentation . . . 190
8.164.2.1 csAttachState . . . 190
8.164.2.2 numRadioInterfaces . . . 190
8.164.2.3 psAttachState . . . 190
8.164.2.4 radioInterface . . . 190
8.164.2.5 regState . . . 190
8.164.2.6 selNetwork . . . 190
8.165 nas_sidNid Struct Reference . . . 190
8.165.1 Detailed Description . . . 190
8.165.2 Field Documentation . . . 190
8.165.2.1 nid . . . 190
8.165.2.2 sid . . . 190
8.166 nas_SignalStrengthTlv Struct Reference . . . 190
8.166.1 Detailed Description . . . 190
8.166.2 Field Documentation . . . 191
8.166.2.1 radioInterface . . . 191
8.166.2.2 signalStrength . . . 191
8.166.2.3 TlvPresent . . . 191
8.167 nas_SLQSSignalStrengthsIndReq Struct Reference . . . 191
8.167.1 Detailed Description . . . 191
8.167.2 Field Documentation . . . 191
8.167.2.1 ecioDelta . . . 191
8.167.2.2 ecioThresholdList . . . 192
8.167.2.3 ecioThresholdListLen . . . 192
8.167.2.4 ioDelta . . . 192
8.167.2.5 lteRsrpDelta . . . 192
8.167.2.6 lteSnrDelta . . . 192

8.167.2.7 rsrqDelta . . . 192
8.167.2.8 rxSignalStrengthDelta . . . 192
8.167.2.9 sinrDelta . . . 192
8.167.2.10 sinrThresholdList . . . 192
8.167.2.11 sinrThresholdListLen . . . 192
8.168 nas_SLQSSignalStrengthsInformation Struct Reference . . . 192
8.168.1 Detailed Description . . . 192
8.168.2 Field Documentation . . . 192
8.168.2.1 ecioInfo . . . 192
8.168.2.2 errorRateInfo . . . 192
8.168.2.3 io . . . 192
8.168.2.4 lteRsrpinfo . . . 193
8.168.2.5 lteSnrinfo . . . 193
8.168.2.6 rsrqInfo . . . 193
8.168.2.7 rxSignalStrengthInfo . . . 193
8.168.2.8 sinr . . . 193
8.169 nas_SLQSSignalStrengthsTlv Struct Reference . . . 193
8.169.1 Detailed Description . . . 193
8.169.2 Field Documentation . . . 193
8.169.2.1 sSLQSSignalStrengthsInfo . . . 193
8.169.2.2 TlvPresent . . . 193
8.170 nas_SrvStatusInfo Struct Reference . . . 193
8.170.1 Detailed Description . . . 193
8.170.2 Field Documentation . . . 194
8.170.2.1 isPrefDataPath . . . 194
8.170.2.2 srvStatus . . . 194
8.171 nas_sysInfoCommon Struct Reference . . . 194
8.171.1 Detailed Description . . . 194
8.171.2 Field Documentation . . . 196
8.171.2.1 isSysForbidden . . . 196
8.171.2.2 isSysForbiddenValid . . . 196
8.171.2.3 roamStatus . . . 196
8.171.2.4 roamStatusValid . . . 196
8.171.2.5 srvCapability . . . 196
8.171.2.6 srvCapabilityValid . . . 196
8.171.2.7 srvDomain . . . 196
8.171.2.8 srvDomainValid . . . 196
8.172 nas_TDSCDMAECIOThresh Struct Reference . . . 196
8.172.1 Detailed Description . . . 196
8.172.2 Field Documentation . . . 196

8.172.2.1 pTDSCDMAECIOThreshList . . . 196
8.172.2.2 TDSCDMAECIOThreshListLen . . . 196
8.173 nas_TDSCDMARSCPThresh Struct Reference . . . 197
8.173.1 Detailed Description . . . 197
8.173.2 Field Documentation . . . 197
8.173.2.1 pTDSCDMARSCPThreshList . . . 197
8.173.2.2 TDSCDMARSCPThreshListLen . . . 197
8.174 nas_TDSCDMARSSIThresh Struct Reference . . . 197
8.174.1 Detailed Description . . . 197
8.174.2 Field Documentation . . . 197
8.174.2.1 pTDSCDMARSSIThreshList . . . 197
8.174.2.2 TDSCDMARSSIThreshListLen . . . 198
8.175 nas_TDSCDMASINRCONFThresh Struct Reference . . . 198
8.175.1 Detailed Description . . . 198
8.175.2 Field Documentation . . . 198
8.175.2.1 pTDSCDMASINRCONFThreshList . . . 198
8.175.2.2 TDSCDMASINRCONFThreshListLen . . . 198
8.176 nas_TDSCDMASINRThresh Struct Reference . . . 198
8.176.1 Detailed Description . . . 198
8.176.2 Field Documentation . . . 198
8.176.2.1 pTDSCDMASINRThreshList . . . 198
8.176.2.2 TDSCDMASINRThreshListLen . . . 199
8.177 nas_timeInfo Struct Reference . . . 199
8.177.1 Detailed Description . . . 199
8.177.2 Field Documentation . . . 200
8.177.2.1 day . . . 200
8.177.2.2 dayLtSavingAdj . . . 200
8.177.2.3 dayOfWeek . . . 200
8.177.2.4 hour . . . 200
8.177.2.5 minute . . . 200
8.177.2.6 month . . . 200
8.177.2.7 radioInterface . . . 200
8.177.2.8 second . . . 200
8.177.2.9 timeZone . . . 200
8.177.2.10 tlvPresent . . . 200
8.177.2.11 year . . . 200
8.178 nas_trueIMSI Struct Reference . . . 200
8.178.1 Detailed Description . . . 201
8.178.2 Field Documentation . . . 201
8.178.2.1 imsiT1112 . . . 201

8.178.2.2 imsiTaddrNum . . . 201
8.178.2.3 imsiTS1 . . . 201
8.178.2.4 imsiTS2 . . . 201
8.178.2.5 mccT . . . 201
8.179 nas_txInfo Struct Reference . . . 201
8.179.1 Detailed Description . . . 201
8.179.2 Field Documentation . . . 202
8.179.2.1 isInTraffic . . . 202
8.179.2.2 txPower . . . 202
8.180 nas_UMTSInfo Struct Reference . . . 202
8.180.1 Detailed Description . . . 202
8.180.2 Field Documentation . . . 203
8.180.2.1 cellID . . . 203
8.180.2.2 ecio . . . 203
8.180.2.3 geranInst . . . 203
8.180.2.4 GeranInstInfo . . . 203
8.180.2.5 lac . . . 203
8.180.2.6 plmn . . . 203
8.180.2.7 psc . . . 203
8.180.2.8 rscp . . . 203
8.180.2.9 uarfcn . . . 203
8.180.2.10 umtsInst . . . 204
8.180.2.11 UMTSInstInfo . . . 204
8.181 nas_UMTSinstInfo Struct Reference . . . 204
8.181.1 Detailed Description . . . 204
8.181.2 Field Documentation . . . 204
8.181.2.1 umtsEcio . . . 204
8.181.2.2 umtsPsc . . . 204
8.181.2.3 umtsRscp . . . 204
8.181.2.4 umtsUarfcn . . . 204
8.182 nas_umtsLTENbrCell Struct Reference . . . 204
8.182.1 Detailed Description . . . 205
8.182.2 Field Documentation . . . 205
8.182.2.1 cellIsTDD . . . 205
8.182.2.2 earfcn . . . 205
8.182.2.3 pci . . . 205
8.182.2.4 rsrp . . . 205
8.182.2.5 rsrq . . . 205
8.182.2.6 srxlev . . . 205
8.183 nas_UniversalTime Struct Reference . . . 205

8.183.1 Detailed Description . . . 206
8.183.2 Field Documentation . . . 206
8.183.2.1 day . . . 206
8.183.2.2 dayOfWeek . . . 206
8.183.2.3 hour . . . 206
8.183.2.4 minute . . . 206
8.183.2.5 month . . . 206
8.183.2.6 second . . . 206
8.183.2.7 year . . . 206
8.184 nas_wcdmaCellInfo Struct Reference . . . 206
8.184.1 Detailed Description . . . 207
8.184.2 Field Documentation . . . 207
8.184.2.1 cpich_ecno . . . 207
8.184.2.2 cpich_rscp . . . 207
8.184.2.3 psc . . . 207
8.184.2.4 srxlev . . . 207
8.185 nas_WCDMAECIOThresh Struct Reference . . . 207
8.185.1 Detailed Description . . . 207
8.185.2 Field Documentation . . . 208
8.185.2.1 pWCDMAECIOThreshList . . . 208
8.185.2.2 WCDMAECIOThreshListLen . . . 208
8.186 nas_WCDMAInfoLTENeighborCell Struct Reference . . . 208
8.186.1 Detailed Description . . . 208
8.186.2 Field Documentation . . . 209
8.186.2.1 UMTSLTENbrCell . . . 209
8.186.2.2 umtsLTENbrCellLen . . . 209
8.186.2.3 wcdmaRRCState . . . 209
8.187 nas_WCDMARSSIThresh Struct Reference . . . 209
8.187.1 Detailed Description . . . 209
8.187.2 Field Documentation . . . 209
8.187.2.1 pWCDMARSSIThreshList . . . 209
8.187.2.2 WCDMARSSIThreshListLen . . . 209
8.188 nas_WCDMASysInfo Struct Reference . . . 209
8.188.1 Detailed Description . . . 210
8.188.2 Field Documentation . . . 212
8.188.2.1 cellId . . . 212
8.188.2.2 cellIdValid . . . 212
8.188.2.3 hsCallStatus . . . 212
8.188.2.4 hsCallStatusValid . . . 212
8.188.2.5 hsInd . . . 212

8.188.2.6 hsIndValid . . . 212
8.188.2.7 lac . . . 212
8.188.2.8 lacValid . . . 212
8.188.2.9 MCC . . . 212
8.188.2.10 MNC . . . 212
8.188.2.11 networkIdValid . . . 212
8.188.2.12 psc . . . 212
8.188.2.13 pscValid . . . 212
8.188.2.14 regRejectInfoValid . . . 213
8.188.2.15 rejCause . . . 213
8.188.2.16 rejectSrvDomain . . . 213
8.188.2.17 sysInfoWCDMA . . . 213
8.189 nas_wcdmaUARFCN Struct Reference . . . 213
8.189.1 Detailed Description . . . 213
8.189.2 Field Documentation . . . 213
8.189.2.1 status . . . 213
8.189.2.2 uarfcn . . . 213
8.190 NASBandPreferenceTlv Struct Reference . . . 213
8.190.1 Field Documentation . . . 213
8.190.1.1 band_pref . . . 213
8.190.1.2 TlvPresent . . . 213
8.191 NASEmergencyModeTlv Struct Reference . . . 213
8.191.1 Field Documentation . . . 214
8.191.1.1 EmerMode . . . 214
8.191.1.2 TlvPresent . . . 214
8.192 NasGetLTECphyCaInfo Struct Reference . . . 214
8.192.1 Field Documentation . . . 214
8.192.1.1 PhyCaAggPcellInfo . . . 214
8.192.1.2 PhyCaAggScellDlBw . . . 214
8.192.1.3 PhyCaAggScellIndex . . . 214
8.192.1.4 PhyCaAggScellIndType . . . 214
8.192.1.5 PhyCaAggScellInfo . . . 214
8.193 NASGWAcqOrderPrefTlv Struct Reference . . . 214
8.193.1 Field Documentation . . . 214
8.193.1.1 GWAcqOrderPref . . . 214
8.193.1.2 TlvPresent . . . 214
8.194 NASLTEBandPreferenceTlv Struct Reference . . . 214
8.194.1 Field Documentation . . . 214
8.194.1.1 LTEBandPref . . . 214
8.194.1.2 TlvPresent . . . 215

8.195 NASLteNasReleaseInfoTlv Struct Reference . . . 215
8.195.1 Field Documentation . . . 215
8.195.1.1 nas_major . . . 215
8.195.1.2 nas_minor . . . 215
8.195.1.3 nas_release . . . 215
8.195.1.4 TlvPresent . . . 215
8.196 NASModePreferenceTlv Struct Reference . . . 215
8.196.1 Field Documentation . . . 215
8.196.1.1 ModePref . . . 215
8.196.1.2 TlvPresent . . . 215
8.197 NASNetSelPreferenceTlv Struct Reference . . . 215
8.197.1 Field Documentation . . . 215
8.197.1.1 NetSelPref . . . 215
8.197.1.2 TlvPresent . . . 215
8.198 NASOTAMessageTlv Struct Reference . . . 215
8.198.1 Field Documentation . . . 216
8.198.1.1 data_buf . . . 216
8.198.1.2 data_len . . . 216
8.198.1.3 message_type . . . 216
8.198.1.4 TlvPresent . . . 216
8.199 NASPhyCaAggPcellInfo Struct Reference . . . 216
8.199.1 Detailed Description . . . 216
8.199.2 Field Documentation . . . 217
8.199.2.1 dl_bw_value . . . 217
8.199.2.2 freq . . . 217
8.199.2.3 iLTEbandValue . . . 217
8.199.2.4 pci . . . 217
8.199.2.5 TlvPresent . . . 217
8.200 NASPhyCaAggScellDlBw Struct Reference . . . 217
8.200.1 Detailed Description . . . 217
8.200.2 Field Documentation . . . 217
8.200.2.1 dl_bw_value . . . 217
8.200.2.2 TlvPresent . . . 217
8.201 NASPhyCaAggScellIndex Struct Reference . . . 217
8.201.1 Detailed Description . . . 217
8.201.2 Field Documentation . . . 218
8.201.2.1 scell_idx . . . 218
8.201.2.2 TlvPresent . . . 218
8.202 NASPhyCaAggScellIndType Struct Reference . . . 218
8.202.1 Detailed Description . . . 218

8.202.2 Field Documentation . . . 218
8.202.2.1 freq . . . 218
8.202.2.2 pci . . . 218
8.202.2.3 scell_state . . . 218
8.202.2.4 TlvPresent . . . 218
8.203 NASPhyCaAggScellInfo Struct Reference . . . 219
8.203.1 Detailed Description . . . 219
8.203.2 Field Documentation . . . 219
8.203.2.1 dl_bw_value . . . 219
8.203.2.2 freq . . . 219
8.203.2.3 iLTEbandValue . . . 219
8.203.2.4 pci . . . 219
8.203.2.5 scell_state . . . 219
8.203.2.6 TlvPresent . . . 220
8.204 NASPRLPreferenceTlv Struct Reference . . . 220
8.204.1 Field Documentation . . . 220
8.204.1.1 PRLPref . . . 220
8.204.1.2 TlvPresent . . . 220
8.205 NASQmiCbkNasSwiOTAMessageInd Struct Reference . . . 220
8.205.1 Field Documentation . . . 220
8.205.1.1 nasRelInfoTlv . . . 220
8.205.1.2 otaMsgTlv . . . 220
8.205.1.3 timeTlv . . . 220
8.206 NASQmiCbkNasSystemSelPrefInd Struct Reference . . . 220
8.206.1 Field Documentation . . . 220
8.206.1.1 BPTlv . . . 220
8.206.1.2 EMTlv . . . 220
8.206.1.3 GWAOPTlv . . . 221
8.206.1.4 LBPTlv . . . 221
8.206.1.5 MPTlv . . . 221
8.206.1.6 NSPTlv . . . 221
8.206.1.7 PRLPTlv . . . 221
8.206.1.8 RPTlv . . . 221
8.206.1.9 SDPTlv . . . 221
8.207 NASRoamPreferenceTlv Struct Reference . . . 221
8.207.1 Field Documentation . . . 221
8.207.1.1 RoamPref . . . 221
8.207.1.2 TlvPresent . . . 221
8.208 NASServDomainPrefTlv Struct Reference . . . 221
8.208.1 Field Documentation . . . 221

8.208.1.1 SrvDomainPref . . . 221
8.208.1.2 TlvPresent . . . 221
8.209 NASServingSystemInfo Struct Reference . . . 221
8.209.1 Detailed Description . . . 222
8.209.2 Field Documentation . . . 222
8.209.2.1 csAttachState . . . 222
8.209.2.2 hdrPersonality . . . 223
8.209.2.3 psAttachState . . . 223
8.209.2.4 radioInterfaceList . . . 223
8.209.2.5 radioInterfaceNo . . . 223
8.209.2.6 registrationState . . . 223
8.209.2.7 selectedNetwork . . . 223
8.210 NASTimeInfoTlv Struct Reference . . . 223
8.210.1 Field Documentation . . . 223
8.210.1.1 time . . . 223
8.210.1.2 TlvPresent . . . 223
8.211 newMTMessageTlv Struct Reference . . . 223
8.211.1 Detailed Description . . . 223
8.211.2 Field Documentation . . . 223
8.211.2.1 MTMessageInfo . . . 223
8.211.2.2 TlvPresent . . . 223
8.212 pack_dms_ActivateAutomatic_t Struct Reference . . . 223
8.212.1 Detailed Description . . . 224
8.212.2 Field Documentation . . . 224
8.212.2.1 actCode . . . 224
8.213 pack_dms_GetCustFeaturesV2_t Struct Reference . . . 224
8.213.1 Detailed Description . . . 224
8.213.2 Field Documentation . . . 224
8.213.2.1 cust_id . . . 224
8.213.2.2 list_type . . . 224
8.213.2.3 Tlvresult . . . 224
8.214 pack_dms_ResetToFactoryDefaults_t Struct Reference . . . 224
8.214.1 Detailed Description . . . 225
8.214.2 Field Documentation . . . 225
8.214.2.1 spc . . . 225
8.215 pack_dms_SetActivationStatusCallback_t Struct Reference . . . 225
8.215.1 Detailed Description . . . 225
8.215.2 Field Documentation . . . 225
8.215.2.1 activationState . . . 225
8.216 pack_dms_SetCrashAction_t Struct Reference . . . 225

8.216.1 Detailed Description . . . 225
8.216.2 Field Documentation . . . 226
8.216.2.1 crashAction . . . 226
8.217pack_dms_SetCustFeature_t Struct Reference . . . 226
8.217.1 Field Documentation . . . 226
8.217.1.1 DHCPRelayEnabled . . . 226
8.217.1.2 DisableIMSI . . . 226
8.217.1.3 GpsEnable . . . 226
8.217.1.4 GPSLPM . . . 226
8.217.1.5 GPSSel . . . 226
8.217.1.6 IPFamSupport . . . 226
8.217.1.7 IsVoiceEnabled . . . 226
8.217.1.8 RMAutoConnect . . . 226
8.217.1.9 SMSSupport . . . 226
8.218pack_dms_SetCustFeaturesV2_t Struct Reference . . . 226
8.218.1 Detailed Description . . . 227
8.218.2 Field Documentation . . . 227
8.218.2.1 cust_id . . . 227
8.218.2.2 cust_value . . . 227
8.218.2.3 Tlvresult . . . 227
8.218.2.4 value_length . . . 227
8.219pack_dms_SetEventReport_t Struct Reference . . . 227
8.219.1 Field Documentation . . . 227
8.219.1.1 mode . . . 227
8.220pack_dms_SetPower_t Struct Reference . . . 227
8.220.1 Field Documentation . . . 227
8.220.1.1 mode . . . 228
8.220.1.2 Tlvresult . . . 228
8.221pack_dms_SetUSBComp_t Struct Reference . . . 228
8.221.1 Field Documentation . . . 228
8.221.1.1 Tlvresult . . . 228
8.221.1.2 USBComp . . . 228
8.222pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 228
8.222.1 Detailed Description . . . 228
8.222.2 Field Documentation . . . 228
8.222.2.1 resetInfoInd . . . 228
8.223pack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 228
8.223.1 Detailed Description . . . 228
8.223.2 Field Documentation . . . 229
8.223.2.1 clear . . . 229

8.224 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 229
8.224.1 Detailed Description . . . 229
8.224.2 Field Documentation . . . 229
8.224.2.1 pDestSMSContent . . . 229
8.224.2.2 pDestSMSNum . . . 229
8.225 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 229
8.225.1 Detailed Description . . . 229
8.225.2 Field Documentation . . . 230
8.225.2.1 manString . . . 230
8.225.2.2 modelString . . . 230
8.225.2.3 plasmaIDString . . . 230
8.225.2.4 swVerString . . . 230
8.226 pack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 230
8.226.1 Detailed Description . . . 230
8.226.2 Field Documentation . . . 230
8.226.2.1 nameString . . . 230
8.226.2.2 versionString . . . 230
8.227 pack_dms_UIMChangePIN_t Struct Reference . . . 230
8.227.1 Detailed Description . . . 231
8.227.2 Field Documentation . . . 231
8.227.2.1 id . . . 231
8.227.2.2 newValue . . . 231
8.227.2.3 oldValue . . . 231
8.228 pack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 231
8.228.1 Detailed Description . . . 231
8.228.2 Field Documentation . . . 232
8.228.2.1 facility . . . 232
8.229 pack_dms_UIMGetICCID_t Struct Reference . . . 232
8.229.1 Detailed Description . . . 232
8.229.2 Field Documentation . . . 232
8.229.2.1 Tlvresult . . . 232
8.230 pack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 232
8.230.1 Detailed Description . . . 232
8.230.2 Field Documentation . . . 233
8.230.2.1 facility . . . 233
8.230.2.2 facilityCk . . . 233
8.230.2.3 facilityState . . . 233
8.231 pack_dms_UIMSetPINProtection_t Struct Reference . . . 233
8.231.1 Detailed Description . . . 233
8.231.2 Field Documentation . . . 233

8.231.2.1 bEnable . . . 233
8.231.2.2 id . . . 233
8.231.2.3 value . . . 233
8.232 pack_dms_UIMUnblockControlKey_t Struct Reference . . . 233
8.232.1 Detailed Description . . . 234
8.232.2 Field Documentation . . . 234
8.232.2.1 facility . . . 234
8.232.2.2 facilityCk . . . 234
8.233 pack_dms_UIMUnblockPIN_t Struct Reference . . . 234
8.233.1 Detailed Description . . . 234
8.233.2 Field Documentation . . . 234
8.233.2.1 id . . . 235
8.233.2.2 newPin . . . 235
8.233.2.3 pukValue . . . 235
8.234 pack_dms_UIMVerifyPIN_t Struct Reference . . . 235
8.234.1 Detailed Description . . . 235
8.234.2 Field Documentation . . . 235
8.234.2.1 id . . . 235
8.234.2.2 value . . . 235
8.235 pack_fms_GetImagesPreference_t Struct Reference . . . 235
8.235.1 Detailed Description . . . 235
8.235.2 Field Documentation . . . 235
8.235.2.1 Tlvresult . . . 236
8.236 pack_fms_GetStoredImages_t Struct Reference . . . 236
8.236.1 Detailed Description . . . 236
8.236.2 Field Documentation . . . 236
8.236.2.1 Tlvresult . . . 236
8.237 pack_fms_SetImagesPreference_t Struct Reference . . . 236
8.237.1 Detailed Description . . . 236
8.237.2 Field Documentation . . . 237
8.237.2.1 bForceDownload . . . 237
8.237.2.2 imageListSize . . . 237
8.237.2.3 modemindex . . . 237
8.237.2.4 pImageList . . . 237
8.237.2.5 Tlvresult . . . 237
8.238 pack_loc_Delete_Assist_Data_t Struct Reference . . . 237
8.238.1 Detailed Description . . . 237
8.238.2 Field Documentation . . . 238
8.238.2.1 pBdsSVInfo . . . 238
8.238.2.2 pCellDb . . . 238

8.238.2.3 pClkInfo . . . 238
8.238.2.4 pGnssData . . . 238
8.238.2.5 pSVInfo . . . 238
8.238.2.6 TlvResult . . . 238
8.239 pack_loc_EventRegister_t Struct Reference . . . 238
8.239.1 Detailed Description . . . 238
8.239.2 Field Documentation . . . 240
8.239.2.1 eventRegister . . . 240
8.239.2.2 TlvResult . . . 240
8.240 pack_loc_SetExtPowerState_t Struct Reference . . . 240
8.240.1 Detailed Description . . . 240
8.240.2 Field Documentation . . . 240
8.240.2.1 extPowerState . . . 240
8.240.2.2 TlvResult . . . 240
8.241 pack_loc_SetOperationMode_t Struct Reference . . . 240
8.241.1 Detailed Description . . . 241
8.241.2 Field Documentation . . . 241
8.241.2.1 mode . . . 241
8.241.2.2 TlvResult . . . 241
8.242 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 241
8.242.1 Detailed Description . . . 241
8.242.2 Field Documentation . . . 241
8.242.2.1 TlvResult . . . 241
8.242.2.2 xid . . . 241
8.243 pack_loc_SLQSLOCInjectPosition_t Struct Reference . . . 241
8.243.1 Detailed Description . . . 242
8.243.2 Field Documentation . . . 245
8.243.2.1 altitudeSrcInfo . . . 245
8.243.2.2 altitudeWrtEllipsoid . . . 245
8.243.2.3 altitudeWrtMeanSeaLevel . . . 245
8.243.2.4 has_altitudeSrcInfo . . . 245
8.243.2.5 has_altitudeWrtEllipsoid . . . 245
8.243.2.6 has_altitudeWrtMeanSeaLevel . . . 245
8.243.2.7 has_horConfidence . . . 245
8.243.2.8 has_horReliability . . . 245
8.243.2.9 has_horUncCircular . . . 246
8.243.2.10 has_latitude . . . 246
8.243.2.11 has_longitude . . . 246
8.243.2.12 has_positionSrc . . . 246
8.243.2.13 has_rawHorConfidence . . . 246

8.243.2.14 has_rawHorUncCircular . . . 246
8.243.2.15 has_timestampAge . . . 246
8.243.2.16 has_timestampUtc . . . 246
8.243.2.17 has_vertConfidence . . . 246
8.243.2.18 has_vertRelicability . . . 246
8.243.2.19 has_vertUnc . . . 246
8.243.2.20 horConfidence . . . 246
8.243.2.21 horReliability . . . 246
8.243.2.22 horUncCircular . . . 246
8.243.2.23 latitude . . . 246
8.243.2.24 longitude . . . 246
8.243.2.25 positionSrc . . . 246
8.243.2.26 rawHorConfidence . . . 246
8.243.2.27 rawHorUncCircular . . . 246
8.243.2.28 timestampAge . . . 246
8.243.2.29 timestampUtc . . . 246
8.243.2.30 vertConfidence . . . 246
8.243.2.31 vertReliability . . . 246
8.243.2.32 vertUnc . . . 246
8.244 pack_loc_SLQSLOCInjectSensorData_t Struct Reference . . . 246
8.244.1 Detailed Description . . . 247
8.244.2 Field Documentation . . . 248
8.244.2.1 acceleroData . . . 248
8.244.2.2 acceleroTimeSrc . . . 248
8.244.2.3 accelTemp . . . 248
8.244.2.4 gyroData . . . 248
8.244.2.5 gyroTemp . . . 248
8.244.2.6 gyroTimeSrc . . . 248
8.244.2.7 has_acceleroTimeSrc . . . 248
8.244.2.8 has_accelTemp . . . 248
8.244.2.9 has_accleroData . . . 248
8.244.2.10 has_gyroData . . . 248
8.244.2.11 has_gyroTemp . . . 248
8.244.2.12 has_gyroTimeSrc . . . 248
8.244.2.13 has_opaqueId . . . 248
8.244.2.14 opaqueId . . . 248
8.245 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference . . . 248
8.245.1 Detailed Description . . . 248
8.245.2 Field Documentation . . . 249
8.245.2.1 timeMsec . . . 249

8.245.2.2 timeUncMsec . . . 249
8.246 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference . . . 249
8.246.1 Detailed Description . . . 249
8.246.2 Field Documentation . . . 249
8.246.2.1 confidence . . . 249
8.246.2.2 has_confidence . . . 249
8.246.2.3 state . . . 249
8.247 pack_loc_Start_t Struct Reference . . . 249
8.247.1 Detailed Description . . . 250
8.247.2 Field Documentation . . . 250
8.247.2.1 pApplicationInfo . . . 251
8.247.2.2 pConfigAltitudeAssumed . . . 251
8.247.2.3 pHorizontalAccuracyLvl . . . 251
8.247.2.4 pIntermediateReportState . . . 251
8.247.2.5 pMinIntervalTime . . . 251
8.247.2.6 pRecurrenceType . . . 251
8.247.2.7 SessionId . . . 251
8.247.2.8 Tlvresult . . . 251
8.248 pack_loc_Stop_t Struct Reference . . . 251
8.248.1 Detailed Description . . . 251
8.248.2 Field Documentation . . . 251
8.248.2.1 SessionId . . . 251
8.248.2.2 Tlvresult . . . 251
8.249 pack_nas_InitiateDomainAttach_t Struct Reference . . . 251
8.249.1 Detailed Description . . . 251
8.249.2 Field Documentation . . . 252
8.249.2.1 action . . . 252
8.250 pack_nas_SetACCOLC_t Struct Reference . . . 252
8.250.1 Detailed Description . . . 252
8.250.2 Field Documentation . . . 252
8.250.2.1 accolc . . . 252
8.250.2.2 spc . . . 252
8.251 pack_nas_SetCDMANetworkParameters_t Struct Reference . . . 252
8.251.1 Detailed Description . . . 252
8.251.2 Field Documentation . . . 253
8.251.2.1 pApplication . . . 253
8.251.2.2 pBroadcast . . . 253
8.251.2.3 pCustomSCP . . . 253
8.251.2.4 pForceRev0 . . . 254
8.251.2.5 pProtocol . . . 254

8.251.2.6 pRoaming . . . 254
8.251.2.7 pSPC . . . 254
8.252 pack_nas_SetNetworkPreference_t Struct Reference . . . 254
8.252.1 Detailed Description . . . 254
8.252.2 Field Documentation . . . 254
8.252.2.1 Duration . . . 254
8.252.2.2 TechnologyPref . . . 254
8.252.2.3 Tlvresult . . . 254
8.253 pack_nas_SLQSConfigSigInfo_t Struct Reference . . . 255
8.253.1 Detailed Description . . . 255
8.253.2 Field Documentation . . . 256
8.253.2.1 pECIOThresh . . . 256
8.253.2.2 pHDRSINRThresh . . . 256
8.253.2.3 pIOThresh . . . 256
8.253.2.4 pLTESigRptCfg . . . 256
8.253.2.5 pLTESNRThresh . . . 256
8.253.2.6 pRSRPThresh . . . 256
8.253.2.7 pRSRQThresh . . . 256
8.253.2.8 pRSSIThresh . . . 256
8.253.2.9 pTDSCDMASINRCONFThresh . . . 256
8.254 pack_nas_SLQSGetPLMNName_t Struct Reference . . . 256
8.254.1 Detailed Description . . . 256
8.254.2 Field Documentation . . . 257
8.254.2.1 mcc . . . 257
8.254.2.2 mnc . . . 257
8.254.2.3 pMncPcsStatus . . . 257
8.255 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference . . . 257
8.255.1 Detailed Description . . . 257
8.255.2 Field Documentation . . . 257
8.255.2.1 pChangeDuration . . . 258
8.255.2.2 pMncPcsDigitStatus . . . 258
8.255.2.3 pMNRInfo . . . 258
8.255.2.4 regAction . . . 258
8.256 pack_nas_SLQSNasConfigSigInfo2_t Struct Reference . . . 258
8.256.1 Detailed Description . . . 259
8.256.2 Field Documentation . . . 261
8.256.2.1 pCDMAECIODelta . . . 261
8.256.2.2 pCDMAECIOThresh . . . 261
8.256.2.3 pCDMARSSIDelta . . . 261
8.256.2.4 pCDMARSSIThresh . . . 261

8.256.2.5 pGSMRSSIDelta . . . 261
8.256.2.6 pGSMRSSIThresh . . . 261
8.256.2.7 pHDRECIODelta . . . 261
8.256.2.8 pHDRECIOThresh . . . 261
8.256.2.9 pHDRIODelta . . . 261
8.256.2.10 pHDRIOThresh . . . 261
8.256.2.11 pHDRRSSIDelta . . . 261
8.256.2.12 pHDRRSSIThresh . . . 261
8.256.2.13 pHDRSINRDelta . . . 261
8.256.2.14 pHDRSINRThresh . . . 261
8.256.2.15 pLTERSRPDelta . . . 261
8.256.2.16 pLTERSRPThresh . . . 261
8.256.2.17 pLTERSRQDelta . . . 262
8.256.2.18 pLTERSRQThresh . . . 262
8.256.2.19 pLTERSSIDelta . . . 262
8.256.2.20 pLTERSSIThresh . . . 262
8.256.2.21 pLTESigRptConfig . . . 262
8.256.2.22 pLTESNRDelta . . . 262
8.256.2.23 pLTESNRThresh . . . 262
8.256.2.24 pTDSCDMAECIODelta . . . 262
8.256.2.25 pTDSCDMAECIOThresh . . . 262
8.256.2.26 pTDSCDMARSCPDelta . . . 262
8.256.2.27 pTDSCDMARSCPThresh . . . 262
8.256.2.28 pTDSCDMARSSIDelta . . . 262
8.256.2.29 pTDSCDMARSSIThresh . . . 262
8.256.2.30 pTDSCDMASINRDelta . . . 262
8.256.2.31 pTDSCDMASINRThresh . . . 262
8.256.2.32 pWCDMAECIODelta . . . 262
8.256.2.33 pWCDMAECIOThresh . . . 262
8.256.2.34 pWCDMARSSIDelta . . . 262
8.256.2.35 pWCDMARSSIThresh . . . 262
8.257 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 262
8.257.1 Detailed Description . . . 262
8.257.2 Field Documentation . . . 263
8.257.2.1 namID . . . 263
8.258 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 263
8.258.1 Detailed Description . . . 263
8.258.2 Field Documentation . . . 263
8.258.2.1 radio_if . . . 263
8.259 pack_nas_SLQSNasIndicationRegisterExt_t Struct Reference . . . 263

8.259.1 Detailed Description . . . 264
8.259.2 Field Documentation . . . 266
8.259.2.1 pDDTMInd . . . 266
8.259.2.2 pDualStandByPrefInd . . . 266
8.259.2.3 pErrorRateInd . . . 266
8.259.2.4 pHDRNewUATIAssInd . . . 266
8.259.2.5 pHDRSessionCloseInd . . . 266
8.259.2.6 pLTECphyCa . . . 266
8.259.2.7 pManagedRoamingInd . . . 266
8.259.2.8 pNetworkRejectInd . . . 266
8.259.2.9 pNetworkTimeInd . . . 266
8.259.2.10 pServingSystemInd . . . 266
8.259.2.11 pSignalStrengthInd . . . 266
8.259.2.12 pSubscriptionInfoInd . . . 266
8.259.2.13 pSuppressSysInfoInd . . . 266
8.259.2.14 pSysInfoInd . . . 266
8.259.2.15 pSystemSelectionInd . . . 266
8.260 pack_nas_SLQSNasSwiIndicationRegister_t Struct Reference . . . 266
8.260.1 Detailed Description . . . 266
8.260.2 Field Documentation . . . 267
8.260.2.1 gsmUmtsDl . . . 267
8.260.2.2 gsmUmtsUl . . . 267
8.260.2.3 lteEmmDl . . . 267
8.260.2.4 lteEmmUl . . . 267
8.260.2.5 lteEsmDl . . . 267
8.260.2.6 lteEsmUl . . . 267
8.260.2.7 pRankIndicatorInd . . . 267
8.260.2.8 pTimer . . . 267
8.261 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference . . . 267
8.261.1 Detailed Description . . . 268
8.261.2 Field Documentation . . . 268
8.261.2.1 pLteEARFCN . . . 268
8.261.2.2 pLtePCI . . . 268
8.261.2.3 pWcdmaUARFCN . . . 268
8.262 pack_nas_SLQSSetBandPreference_t Struct Reference . . . 268
8.262.1 Detailed Description . . . 268
8.262.2 Field Documentation . . . 269
8.262.2.1 bandPref . . . 270
8.263 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference . . . 270
8.263.1 Detailed Description . . . 270

8.263.2 Field Documentation . . . 270
8.263.2.1 bEnable . . . 270
8.263.2.2 pSigIndReq . . . 270
8.264 pack_nas_SLQSSetSysSelectionPref_t Struct Reference . . . 270
8.264.1 Detailed Description . . . 270
8.264.2 Field Documentation . . . 274
8.264.2.1 pAcqOrderPref . . . 274
8.264.2.2 pBandPref . . . 274
8.264.2.3 pChgDuration . . . 274
8.264.2.4 pCSGID . . . 274
8.264.2.5 pEmerMode . . . 274
8.264.2.6 pGWAcqOrderPref . . . 274
8.264.2.7 pLTEBandPref . . . 275
8.264.2.8 pMNCIncPCSDigStat . . . 275
8.264.2.9 pModePref . . . 275
8.264.2.10 pNetSelPref . . . 275
8.264.2.11 pPRLPref . . . 275
8.264.2.12 pRAT . . . 275
8.264.2.13 pRoamPref . . . 275
8.264.2.14 pSrvDomainPref . . . 275
8.264.2.15 pSrvRegRestriction . . . 275
8.264.2.16 pTdscdmaBandPref . . . 275
8.265 pack_nas_SLQSSwiPSDetach_t Struct Reference . . . 275
8.265.1 Detailed Description . . . 275
8.265.2 Field Documentation . . . 275
8.265.2.1 pDetachAction . . . 275
8.266 pack_pds_SetPDSDefaults_t Struct Reference . . . 275
8.266.1 Detailed Description . . . 275
8.266.2 Field Documentation . . . 276
8.266.2.1 accuracy . . . 276
8.266.2.2 interval . . . 276
8.266.2.3 operation . . . 276
8.266.2.4 timeout . . . 276
8.267 pack_pds_SetPDSState_t Struct Reference . . . 276
8.267.1 Detailed Description . . . 276
8.267.2 Field Documentation . . . 276
8.267.2.1 enable . . . 276
8.268 pack_pds_SetPortAutomaticTracking_t Struct Reference . . . 276
8.268.1 Detailed Description . . . 277
8.268.2 Field Documentation . . . 277

8.268.2.1 bAuto . . . 277
8.269 pack_qmi_t Struct Reference . . . 277
8.269.1 Detailed Description . . . 277
8.269.2 Field Documentation . . . 277
8.269.2.1 msgid . . . 277
8.269.2.2 svc . . . 277
8.269.2.3 timeout . . . 277
8.269.2.4 xid . . . 277
8.270 pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 277
8.270.1 Detailed Description . . . 278
8.270.2 Field Documentation . . . 278
8.270.2.1 apnId . . . 278
8.271 pack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 278
8.271.1 Detailed Description . . . 278
8.271.2 Field Documentation . . . 278
8.271.2.1 apnId . . . 278
8.272 pack_qos_SLQSSetQosEventCallback_t Struct Reference . . . 278
8.272.1 Detailed Description . . . 278
8.272.2 Field Documentation . . . 279
8.272.2.1 enable . . . 279
8.273 pack_rms_SetSMSWake_t Struct Reference . . . 279
8.273.1 Detailed Description . . . 279
8.273.2 Field Documentation . . . 279
8.273.2.1 enabled . . . 279
8.273.2.2 wake_mask . . . 279
8.274 pack_sar_SLQSSetRfSarState_t Struct Reference . . . 279
8.274.1 Detailed Description . . . 279
8.274.2 Field Documentation . . . 280
8.274.2.1 RfSarState . . . 280
8.275 pack_sms_SaveSMS_t Struct Reference . . . 280
8.275.1 Detailed Description . . . 280
8.275.2 Field Documentation . . . 281
8.275.2.1 messageFormat . . . 281
8.275.2.2 messageSize . . . 281
8.275.2.3 pMessage . . . 281
8.275.2.4 storageType . . . 281
8.276 pack_sms_SendSMS_t Struct Reference . . . 281
8.276.1 Detailed Description . . . 281
8.276.2 Field Documentation . . . 281
8.276.2.1 messageFormat . . . 281

8.276.2.2 messageSize . . . 281
8.276.2.3 pLinktimer . . . 281
8.276.2.4 pMessage . . . 281
8.277 pack_sms_SetNewSMSCallback_t Struct Reference . . . 281
8.277.1 Detailed Description . . . 282
8.277.2 Field Documentation . . . 282
8.277.2.1 status . . . 282
8.278 pack_sms_SetSMSCAddress_t Struct Reference . . . 282
8.278.1 Detailed Description . . . 282
8.278.2 Field Documentation . . . 282
8.278.2.1 pSMSCAddress . . . 282
8.278.2.2 pSMSCType . . . 282
8.279 pack_sms_SLQSDeleteSMS_t Struct Reference . . . 282
8.279.1 Detailed Description . . . 282
8.279.2 Field Documentation . . . 283
8.279.2.1 pMessageIndex . . . 283
8.279.2.2 pMessageMode . . . 283
8.279.2.3 pMessageTag . . . 283
8.279.2.4 storageType . . . 283
8.280 pack_sms_SLQSGetSMS_t Struct Reference . . . 283
8.280.1 Detailed Description . . . 283
8.280.2 Field Documentation . . . 284
8.280.2.1 messageIndex . . . 284
8.280.2.2 pMessageMode . . . 284
8.280.2.3 storageType . . . 284
8.281 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 284
8.281.1 Detailed Description . . . 284
8.281.2 Field Documentation . . . 284
8.281.2.1 mode . . . 284
8.282 pack_sms_SLQSGetSMSList_t Struct Reference . . . 284
8.282.1 Detailed Description . . . 284
8.282.2 Field Documentation . . . 285
8.282.2.1 pMessageMode . . . 285
8.282.2.2 pRequestedTag . . . 285
8.282.2.3 storageType . . . 285
8.283 pack_sms_SLQSModifySMSStatus_t Struct Reference . . . 285
8.283.1 Detailed Description . . . 285
8.283.2 Field Documentation . . . 286
8.283.2.1 messageIndex . . . 286
8.283.2.2 messageTag . . . 286

8.283.2.3 pMessageMode . . . 286
8.283.2.4 storageType . . . 286
8.284 pack_sms_SLQSSendAsyncSMS_t Struct Reference . . . 286
8.284.1 Detailed Description . . . 286
8.284.2 Field Documentation . . . 286
8.284.2.1 pSendSmsParams . . . 286
8.285 pack_sms_SLQSSetIndicationRegister_t Struct Reference . . . 286
8.285.1 Detailed Description . . . 286
8.285.2 Field Documentation . . . 287
8.285.2.1 pSetIndicationRegReq . . . 287
8.286 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference . . . 287
8.286.1 Detailed Description . . . 287
8.286.2 Field Documentation . . . 287
8.286.2.1 broadcastActivate . . . 287
8.286.2.2 mode . . . 287
8.287 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference . . . 287
8.287.1 Detailed Description . . . 287
8.287.2 Field Documentation . . . 288
8.287.2.1 mode . . . 288
8.287.2.2 pBroadcastConfig . . . 288
8.287.2.3 pCDMABroadcastConfig . . . 288
8.288 pack_sms_SLQSSetSmsStorage_t Struct Reference . . . 288
8.288.1 Detailed Description . . . 288
8.288.2 Field Documentation . . . 288
8.288.2.1 smsStorage . . . 288
8.289 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 288
8.289.1 Detailed Description . . . 288
8.289.2 Field Documentation . . . 289
8.289.2.1 pMaxStorageSizeReq . . . 289
8.290 pack_sms_SLQSSmsSetRoutes_t Struct Reference . . . 289
8.290.1 Detailed Description . . . 289
8.290.2 Field Documentation . . . 289
8.290.2.1 pSetRoutesReq . . . 289
8.291 pack_swiavms_SLQSAvMSSendSelection_t Struct Reference . . . 289
8.291.1 Detailed Description . . . 289
8.291.2 Field Documentation . . . 290
8.291.2.1 pClientPerformOperationFlag . . . 290
8.291.2.2 pDeferTime . . . 290
8.291.2.3 pPackageID . . . 290
8.291.2.4 pRejectReason . . . 290

8.291.2.5 selection . . . 290
8.292 pack_swiavms_SLQSAvMSSetSettings_t Struct Reference . . . 290
8.292.1 Detailed Description . . . 290
8.292.2 Field Documentation . . . 291
8.292.2.1 AutoConnect . . . 291
8.292.2.2 AutoReboot . . . 291
8.292.2.3 FwAutodownload . . . 291
8.292.2.4 FwAutoUpdate . . . 291
8.292.2.5 pAPNInfo . . . 291
8.292.2.6 pConnectionRetryTimers . . . 292
8.292.2.7 pFwAutoSDM . . . 292
8.292.2.8 pNotifStore . . . 292
8.292.2.9 pPeriodInfo . . . 292
8.292.2.10 pPollingTimer . . . 292
8.293 pack_swiavms_SLQSAvMSStartSession_t Struct Reference . . . 292
8.293.1 Detailed Description . . . 292
8.293.2 Field Documentation . . . 292
8.293.2.1 sessionType . . . 292
8.294 pack_swiavms_SLQSAvMSStopSession_t Struct Reference . . . 292
8.294.1 Detailed Description . . . 292
8.294.2 Field Documentation . . . 293
8.294.2.1 sessionType . . . 293
8.295 pack_swiloc_SwiLocSetAutoStart_t Struct Reference . . . 293
8.295.1 Detailed Description . . . 293
8.295.2 Field Documentation . . . 294
8.295.2.1 fix_rate . . . 294
8.295.2.2 fix_type . . . 294
8.295.2.3 function . . . 294
8.295.2.4 max_dist . . . 294
8.295.2.5 max_time . . . 294
8.295.2.6 set_fix_rate . . . 294
8.295.2.7 set_fix_type . . . 294
8.295.2.8 set_function . . . 294
8.295.2.9 set_max_dist . . . 294
8.295.2.10 set_max_time . . . 294
8.296 pack_swioma_SLQSOMADMCancelSession_t Struct Reference . . . 294
8.296.1 Detailed Description . . . 294
8.296.2 Field Documentation . . . 295
8.296.2.1 sessionType . . . 295
8.297 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 295

8.297.1 Detailed Description . . . 295
8.297.2 Field Documentation . . . 295
8.297.2.1 SessionType . . . 295
8.298 pack_swioma_SLQSOMADMSendSelection_t Struct Reference . . . 295
8.298.1 Detailed Description . . . 295
8.298.2 Field Documentation . . . 296
8.298.2.1 pDeferTime . . . 296
8.298.2.2 pRejectReason . . . 296
8.298.2.3 selection . . . 296
8.299 pack_swioma_SLQSOMADMSetSettings_t Struct Reference . . . 296
8.299.1 Detailed Description . . . 296
8.299.2 Field Documentation . . . 297
8.299.2.1 FOTAdownload . . . 297
8.299.2.2 FOTAUpdate . . . 297
8.299.2.3 pAutosdm . . . 297
8.299.2.4 pFwAutoCheck . . . 297
8.300 pack_swioma_SLQSOMADMStartSession_t Struct Reference . . . 297
8.300.1 Detailed Description . . . 297
8.300.2 Field Documentation . . . 297
8.300.2.1 sessionType . . . 297
8.301 pack_uim_ChangePin_t Struct Reference . . . 297
8.301.1 Detailed Description . . . 298
8.301.2 Field Documentation . . . 298
8.301.2.1 changePIN . . . 298
8.301.2.2 EncryptedPIN1 . . . 298
8.301.2.3 pIndicationToken . . . 298
8.301.2.4 pKeyReferenceID . . . 298
8.301.2.5 sessionInfo . . . 298
8.301.2.6 Tlvresult . . . 298
8.302 pack_uim_ReadTransparent_t Struct Reference . . . 298
8.302.1 Detailed Description . . . 299
8.302.2 Field Documentation . . . 299
8.302.2.1 fileIndex . . . 299
8.302.2.2 pEncryptData . . . 299
8.302.2.3 pIndicationToken . . . 299
8.302.2.4 readTransparent . . . 299
8.302.2.5 sessionInfo . . . 299
8.302.2.6 Tlvresult . . . 299
8.303 pack_uim_SetPinProtection_t Struct Reference . . . 300
8.303.1 Detailed Description . . . 300

8.303.2 Field Documentation . . . 300
8.303.2.1 EncryptedPIN1 . . . 300
8.303.2.2 pIndicationToken . . . 300
8.303.2.3 pinProtection . . . 300
8.303.2.4 pKeyReferenceID . . . 300
8.303.2.5 sessionInfo . . . 300
8.303.2.6 Tlvresult . . . 301
8.304 pack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 301
8.304.1 Detailed Description . . . 301
8.304.2 Field Documentation . . . 301
8.304.2.1 authData . . . 301
8.304.2.2 pIndicationToken . . . 301
8.304.2.3 sessionInfo . . . 301
8.305 pack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 301
8.305.1 Detailed Description . . . 301
8.305.2 Field Documentation . . . 302
8.305.2.1 depersonilisationInfo . . . 302
8.306 pack_uim_SLQSUIMEventRegister_t Struct Reference . . . 302
8.306.1 Detailed Description . . . 302
8.306.2 Field Documentation . . . 302
8.306.2.1 eventMask . . . 302
8.307 pack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 302
8.307.1 Detailed Description . . . 302
8.307.2 Field Documentation . . . 302
8.307.2.1 pConfigurationMask . . . 302
8.308 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 303
8.308.1 Detailed Description . . . 303
8.308.2 Field Documentation . . . 303
8.308.2.1 fileIndex . . . 303
8.308.2.2 pIndicationToken . . . 303
8.308.2.3 sessionInfo . . . 303
8.309 pack_uim_SLQSUIMPowerDown_t Struct Reference . . . 303
8.309.1 Detailed Description . . . 303
8.309.2 Field Documentation . . . 304
8.309.2.1 slot . . . 304
8.310 pack_uim_SLQSUIMPowerUp_t Struct Reference . . . 304
8.310.1 Detailed Description . . . 304
8.310.2 Field Documentation . . . 304
8.310.2.1 pIgnoreHotSwapSwitch . . . 304
8.310.2.2 slot . . . 304

8.311 pack_uim_SLQSUIMRefreshComplete_t Struct Reference . . . 304
8.311.1 Detailed Description . . . 304
8.311.2 Field Documentation . . . 305
8.311.2.1 refreshComplete . . . 305
8.311.2.2 sessionInfo . . . 305
8.312 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 305
8.312.1 Detailed Description . . . 305
8.312.2 Field Documentation . . . 305
8.312.2.1 sessionInfo . . . 305
8.313 pack_uim_SLQSUIMRefreshOK_t Struct Reference . . . 305
8.313.1 Detailed Description . . . 305
8.313.2 Field Documentation . . . 306
8.313.2.1 OKtoRefresh . . . 306
8.313.2.2 sessionInfo . . . 306
8.314 pack_uim_SLQSUIMRefreshRegister_t Struct Reference . . . 306
8.314.1 Detailed Description . . . 306
8.314.2 Field Documentation . . . 306
8.314.2.1 regRefresh . . . 306
8.314.2.2 sessionInfo . . . 306
8.315 pack_uim_SLQSUIMSwitchSlot_t Struct Reference . . . 306
8.315.1 Detailed Description . . . 306
8.315.2 Field Documentation . . . 307
8.315.2.1 bLogicalSlot . . . 307
8.315.2.2 ulPhysicalSlot . . . 307
8.316 pack_uim_UnblockPin_t Struct Reference . . . 307
8.316.1 Detailed Description . . . 307
8.316.2 Field Documentation . . . 308
8.316.2.1 EncryptedPIN1 . . . 308
8.316.2.2 pIndicationToken . . . 308
8.316.2.3 pinProtection . . . 308
8.316.2.4 pKeyReferenceID . . . 308
8.316.2.5 sessionInfo . . . 308
8.316.2.6 Tlvresult . . . 308
8.317 pack_uim_VerifyPin_t Struct Reference . . . 308
8.317.1 Detailed Description . . . 308
8.317.2 Field Documentation . . . 309
8.317.2.1 pEncryptedPIN1 . . . 309
8.317.2.2 pIndicationToken . . . 309
8.317.2.3 pKeyReferenceID . . . 309
8.317.2.4 sessionInfo . . . 309

8.317.2.5 TlvresultTlvresult 309
8.317.2.6 verifyPINverifyPIN 309
8.318pack_voice_AnswerUSSD_t Struct Reference 309
8.318.1 Detailed Description 309
8.318.2 Field Documentation 310
8.318.2.1 pInfo 310
8.319pack_voice_OriginateUSSD_t Struct Reference 310
8.319.1 Detailed Description 310
8.319.2 Field Documentation 310
8.319.2.1 pInfo 310
8.320pack_voice_SLQSOriginateUSSD_t Struct Reference 310
8.320.1 Detailed Description 310
8.320.2 Field Documentation 310
8.320.2.1 ussData 311
8.320.2.2 ussDCS 311
8.320.2.3 ussLen 311
8.321pack_voice_SLQSVoiceALSSelectLine_t Struct Reference 311
8.321.1 Detailed Description 311
8.321.2 Field Documentation 311
8.321.2.1 lineValue 311
8.322pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference 311
8.322.1 Detailed Description 311
8.322.2 Field Documentation 311
8.322.2.1 switchOption 311
8.323pack_voice_SLQSVoiceAnswerCall_t Struct Reference 311
8.323.1 Detailed Description 312
8.323.2 Field Documentation 312
8.323.2.1 pCallId 312
8.324pack_voice_SLQSVoiceBindSubscription_t Struct Reference 312
8.324.1 Detailed Description 312
8.324.2 Field Documentation 312
8.324.2.1 subsType 312
8.325pack_voice_SLQSVoiceBurstDTMF_t Struct Reference 312
8.325.1 Detailed Description 312
8.325.2 Field Documentation 313
8.325.2.1 BurstDTMFInfo 313
8.325.2.2 pBurstDTMFLengths 313
8.326pack_voice_SLQSVoiceDialCall_t Struct Reference 313
8.326.1 Detailed Description 313
8.326.2 Field Documentation 314

8.326.2.1 callNumber . . . 314
8.326.2.2 pCallPartySubAdd . . . 314
8.326.2.3 pCallType . . . 314
8.326.2.4 pCLIRType . . . 314
8.326.2.5 pCUGInfo . . . 314
8.326.2.6 pEmergencyCategory . . . 314
8.326.2.7 pSvcType . . . 314
8.326.2.8 pUUSInFo . . . 314
8.327 pack_voice_SLQSVoiceEndCall_t Struct Reference . . . 315
8.327.1 Detailed Description . . . 315
8.327.2 Field Documentation . . . 315
8.327.2.1 pCallId . . . 315
8.328 pack_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 315
8.328.1 Detailed Description . . . 315
8.328.2 Field Documentation . . . 316
8.328.2.1 pSvcClass . . . 316
8.328.2.2 reason . . . 316
8.329 pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 316
8.329.1 Detailed Description . . . 316
8.329.2 Field Documentation . . . 316
8.329.2.1 pSvcClass . . . 316
8.329.2.2 Reason . . . 316
8.330 pack_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 316
8.330.1 Detailed Description . . . 317
8.330.2 Field Documentation . . . 317
8.330.2.1 callID . . . 317
8.331 pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 317
8.331.1 Detailed Description . . . 317
8.331.2 Field Documentation . . . 317
8.331.2.1 pSvcClass . . . 317
8.332 pack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 317
8.332.1 Detailed Description . . . 318
8.332.2 Field Documentation . . . 319
8.332.2.1 pAirTimer . . . 319
8.332.2.2 pAMRStatus . . . 319
8.332.2.3 pAutoAnswer . . . 319
8.332.2.4 pNamID . . . 319
8.332.2.5 pPrefVoicePrivacy . . . 319
8.332.2.6 pPrefVoiceSO . . . 319
8.332.2.7 pRoamTimer . . . 319

8.332.2.8 pTTYMode . . . 319
8.332.2.9 pVoiceDomainPref . . . 319
8.333 pack_voice_SLQSVoiceIndicationRegister_t Struct Reference . . . 319
8.333.1 Detailed Description . . . 319
8.333.2 Field Documentation . . . 320
8.333.2.1 pRegDTMFEvents . . . 320
8.333.2.2 pRegVoicePrivacyEvents . . . 320
8.333.2.3 pSuppsNotifEvents . . . 320
8.334 pack_voice_SLQSVoiceManageCalls_t Struct Reference . . . 320
8.334.1 Detailed Description . . . 320
8.334.2 Field Documentation . . . 320
8.334.2.1 pCallID . . . 321
8.334.2.2 SUPSType . . . 321
8.335 pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference . . . 321
8.335.1 Detailed Description . . . 321
8.335.2 Field Documentation . . . 321
8.335.2.1 USSInformation . . . 321
8.336 pack_voice_SLQSVoiceSendFlash_t Struct Reference . . . 321
8.336.1 Detailed Description . . . 321
8.336.2 Field Documentation . . . 322
8.336.2.1 pCallID . . . 322
8.336.2.2 pFlashPayLd . . . 322
8.336.2.3 pFlashType . . . 322
8.337 pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 322
8.337.1 Detailed Description . . . 322
8.337.2 Field Documentation . . . 323
8.337.2.1 newPasswd . . . 323
8.337.2.2 newPasswdAgain . . . 323
8.337.2.3 oldPasswd . . . 323
8.337.2.4 Reason . . . 323
8.338 pack_voice_SLQSVoiceSetConfig_t Struct Reference . . . 323
8.338.1 Detailed Description . . . 323
8.338.2 Field Documentation . . . 324
8.338.2.1 pAirTimerConfig . . . 324
8.338.2.2 pAutoAnswer . . . 324
8.338.2.3 pPrefVoiceDomain . . . 324
8.338.2.4 pPrefVoiceSO . . . 324
8.338.2.5 pRoamTimerConfig . . . 324
8.338.2.6 pTTYMode . . . 324
8.339 pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference . . . 324

8.339.1 Detailed Description . . . 324
8.339.2 Field Documentation . . . 324
8.339.2.1 privacyPref . . . 324
8.340 pack_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 324
8.340.1 Detailed Description . . . 325
8.340.2 Field Documentation . . . 327
8.340.2.1 pCallBarringPasswd . . . 327
8.340.2.2 pCallForwardingNumber . . . 327
8.340.2.3 pCallFwdTypeAndPlan . . . 327
8.340.2.4 pServiceClass . . . 327
8.340.2.5 pTimerVal . . . 327
8.340.2.6 reason . . . 327
8.340.2.7 voiceSvc . . . 327
8.341 pack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 327
8.341.1 Detailed Description . . . 327
8.341.2 Field Documentation . . . 327
8.341.2.1 DTMFdigit . . . 327
8.341.2.2 pCallID . . . 328
8.342 pack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 328
8.342.1 Detailed Description . . . 328
8.342.2 Field Documentation . . . 328
8.342.2.1 callID . . . 328
8.343 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference . . . 328
8.343.1 Detailed Description . . . 328
8.343.2 Field Documentation . . . 328
8.343.2.1 pEnableNotification . . . 328
8.344 pack_wds_GetDefaultProfile_t Struct Reference . . . 328
8.344.1 Detailed Description . . . 329
8.344.2 Field Documentation . . . 329
8.344.2.1 profiletype . . . 329
8.345 pack_wds_GetDefaultProfileNum_t Struct Reference . . . 329
8.345.1 Detailed Description . . . 329
8.345.2 Field Documentation . . . 329
8.345.2.1 family . . . 329
8.345.2.2 type . . . 329
8.346 pack_wds_GetDormancyState_t Struct Reference . . . 329
8.347 pack_wds_GetLastMobileIPError_t Struct Reference . . . 329
8.348 pack_wds_GetMobileIP_t Struct Reference . . . 329
8.349 pack_wds_GetMobileIPProfile_t Struct Reference . . . 329
8.349.1 Detailed Description . . . 329

8.349.2 Field Documentation . . . 330
8.349.2.1 index . . . 330
8.350 pack_wds_GetPacketStatistics_t Struct Reference . . . 330
8.350.1 Detailed Description . . . 330
8.350.2 Field Documentation . . . 330
8.350.2.1 pStatMask . . . 330
8.351 pack_wds_GetPacketStatus_t Struct Reference . . . 330
8.351.1 Detailed Description . . . 330
8.351.2 Field Documentation . . . 330
8.351.2.1 statmask . . . 330
8.352 pack_wds_GetSessionDuration_t Struct Reference . . . 330
8.353 pack_wds_RMSetTransferStatistics_t Struct Reference . . . 330
8.353.1 Detailed Description . . . 331
8.353.2 Field Documentation . . . 331
8.353.2.1 RmTrasnferStaticsReq . . . 331
8.354 pack_wds_SetAutoconnect_t Struct Reference . . . 331
8.354.1 Detailed Description . . . 331
8.354.2 Field Documentation . . . 331
8.354.2.1 acroamsetting . . . 331
8.354.2.2 acsetting . . . 331
8.355 pack_wds_SetDefaultProfile_t Struct Reference . . . 331
8.355.1 Detailed Description . . . 331
8.355.2 Field Documentation . . . 332
8.355.2.1 authentication . . . 332
8.355.2.2 ipAddress . . . 332
8.355.2.3 pApnname . . . 332
8.355.2.4 pdpType . . . 332
8.355.2.5 pName . . . 332
8.355.2.6 pPassword . . . 332
8.355.2.7 primaryDNS . . . 332
8.355.2.8 profileType . . . 332
8.355.2.9 pUsername . . . 332
8.355.2.10 secondaryDNS . . . 332
8.356 pack_wds_SetDefaultProfileNum_t Struct Reference . . . 332
8.356.1 Field Documentation . . . 332
8.356.1.1 family . . . 332
8.356.1.2 index . . . 332
8.356.1.3 type . . . 332
8.357 pack_wds_SetMobileIP_t Struct Reference . . . 332
8.357.1 Detailed Description . . . 333

8.357.2 Field Documentation . . . 333
8.357.2.1 mode . . . 333
8.358 pack_wds_SetMobileIPParameters_t Struct Reference . . . 333
8.358.1 Detailed Description . . . 333
8.358.2 Field Documentation . . . 334
8.358.2.1 pHA2002bis . . . 334
8.358.2.2 pHAAuthenticator . . . 334
8.358.2.3 pMode . . . 334
8.358.2.4 pReRegPeriod . . . 334
8.358.2.5 pReRegTraffic . . . 334
8.358.2.6 pRetryInterval . . . 334
8.358.2.7 pRetryLimit . . . 334
8.358.2.8 pSPC . . . 334
8.359 pack_wds_SetMobileIPProfile_t Struct Reference . . . 334
8.359.1 Detailed Description . . . 335
8.359.2 Field Documentation . . . 335
8.359.2.1 index . . . 335
8.359.2.2 pAAASPI . . . 335
8.359.2.3 pAddress . . . 335
8.359.2.4 pEnabled . . . 335
8.359.2.5 pHASPI . . . 335
8.359.2.6 pMNAAA . . . 335
8.359.2.7 pMNHA . . . 335
8.359.2.8 pNAI . . . 335
8.359.2.9 pPrimaryHA . . . 335
8.359.2.10 pRevTunneling . . . 335
8.359.2.11 pSecondaryHA . . . 335
8.359.2.12 spc . . . 335
8.360 pack_wds_SLQSCreateProfile_t Struct Reference . . . 335
8.360.1 Detailed Description . . . 335
8.360.2 Field Documentation . . . 336
8.360.2.1 pCurProfile . . . 336
8.360.2.2 pProfileId . . . 336
8.360.2.3 pProfileType . . . 336
8.361 pack_wds_SLQSDeleteProfile_t Struct Reference . . . 336
8.361.1 Detailed Description . . . 336
8.361.2 Field Documentation . . . 336
8.361.2.1 profileIndex . . . 336
8.361.2.2 profileType . . . 336
8.362 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 336

8.363pack_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 337
8.364pack_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 337
8.364.1 Detailed Description . . . 337
8.364.2 Field Documentation . . . 337
8.364.2.1 Mask . . . 337
8.364.2.2 pReportChannelRate . . . 337
8.364.2.3 pReportConnStatus . . . 337
8.364.2.4 pReportDataBearerTech . . . 337
8.364.2.5 pReportDormStatus . . . 337
8.364.2.6 pTransferStatInd . . . 337
8.365pack_wds_SLQSGetProfileSettings_t Struct Reference . . . 337
8.365.1 Detailed Description . . . 337
8.365.2 Field Documentation . . . 338
8.365.2.1 ProfileId . . . 338
8.365.2.2 ProfileType . . . 338
8.366pack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 338
8.366.1 Detailed Description . . . 338
8.366.2 Field Documentation . . . 339
8.366.2.1 pReqSettings . . . 339
8.367pack_wds_SLQSModifyProfile_t Struct Reference . . . 339
8.367.1 Detailed Description . . . 339
8.367.2 Field Documentation . . . 340
8.367.2.1 curProfile . . . 340
8.367.2.2 pProfileId . . . 340
8.367.2.3 pProfileType . . . 340
8.368pack_wds_SLQSSet3GPPConfigItem_t Struct Reference . . . 340
8.368.1 Detailed Description . . . 340
8.368.2 Field Documentation . . . 341
8.368.2.1 LTEAttachProfileListLen . . . 341
8.368.2.2 p3gppRelease . . . 341
8.368.2.3 pDefaultPDNEnabled . . . 341
8.368.2.4 pLTEAttachProfile . . . 341
8.368.2.5 pLTEAttachProfileList . . . 341
8.368.2.6 pProfileList . . . 342
8.369pack_wds_SLQSSetIPFamilyPreference_t Struct Reference . . . 342
8.369.1 Detailed Description . . . 342
8.369.2 Field Documentation . . . 342
8.369.2.1 IPFamilyPreference . . . 342
8.370pack_wds_SLQSSetWdsEventCallback_t Struct Reference . . . 342
8.370.1 Detailed Description . . . 342

8.370.2 Field Documentation . . . 342
8.370.2.1 currentDataBearer . . . 342
8.370.2.2 dataBearer . . . 343
8.370.2.3 dataBearerTechExt . . . 343
8.370.2.4 dataSystemStatus . . . 343
8.370.2.5 dormancyStatus . . . 343
8.370.2.6 interval . . . 343
8.370.2.7 mobileIP . . . 343
8.370.2.8 transferStats . . . 343
8.371 pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 343
8.371.1 Detailed Description . . . 343
8.371.2 Field Documentation . . . 343
8.371.2.1 pProfileId . . . 343
8.372 pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference . . . 343
8.372.1 Detailed Description . . . 343
8.372.2 Field Documentation . . . 344
8.372.2.1 pHwConfig . . . 344
8.372.2.2 pProfileId . . . 344
8.372.2.3 pRequestOptionList . . . 344
8.373 pack_wds_SLQSSSetLoopback_t Struct Reference . . . 344
8.373.1 Detailed Description . . . 344
8.373.2 Field Documentation . . . 344
8.373.2.1 loopbackMode . . . 344
8.373.2.2 loopbackMultiplier . . . 344
8.374 pack_wds_SLQSStartDataSession_t Struct Reference . . . 344
8.374.1 Detailed Description . . . 344
8.374.2 Field Documentation . . . 345
8.374.2.1 pAuth . . . 345
8.374.2.2 pPass . . . 345
8.374.2.3 pprofileid3gpp . . . 345
8.374.2.4 pprofileid3gpp2 . . . 345
8.374.2.5 pTech . . . 345
8.374.2.6 pUser . . . 345
8.375 pack_wds_SLQSStopDataSession_t Struct Reference . . . 345
8.375.1 Detailed Description . . . 346
8.375.2 Field Documentation . . . 346
8.375.2.1 psid . . . 346
8.376 pack_wds_SLQSWdsSetEventReport_t Struct Reference . . . 346
8.376.1 Detailed Description . . . 346
8.376.2 Field Documentation . . . 347

8.376.2.1 pCurrChannelRateInd . . . 347
8.376.2.2 pCurrDataBearerTechInd . . . 347
8.376.2.3 pCurrPrefDataSysInd . . . 347
8.376.2.4 pDataBearerTechInd . . . 347
8.376.2.5 pDataCallStatusChangeInd . . . 347
8.376.2.6 pDataSystemStatusChangeInd . . . 347
8.376.2.7 pDormancyStatusInd . . . 347
8.376.2.8 pEVDOPageMonPerChangeInd . . . 347
8.376.2.9 pMIPStatusInd . . . 347
8.376.2.10 pTransferStatInd . . . 347
8.377 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 348
8.377.1 Detailed Description . . . 348
8.377.2 Field Documentation . . . 348
8.377.2.1 contextId . . . 348
8.377.2.2 contextType . . . 348
8.378 PackCreateProfileOut Struct Reference . . . 348
8.378.1 Field Documentation . . . 348
8.378.1.1 ExtErrorCode . . . 348
8.378.1.2 ProfileIndex . . . 348
8.378.1.3 ProfileType . . . 348
8.379 packgetDyingGaspCfg Struct Reference . . . 348
8.379.1 Detailed Description . . . 348
8.379.2 Field Documentation . . . 349
8.379.2.1 pDestSMSContent . . . 349
8.379.2.2 pDestSMSNum . . . 349
8.380 packgetDyingGaspStatistics Struct Reference . . . 349
8.380.1 Detailed Description . . . 349
8.380.2 Field Documentation . . . 349
8.380.2.1 pSMSAttemptedFlag . . . 349
8.380.2.2 pTimeStamp . . . 349
8.381 qmiSmsMessageList Struct Reference . . . 349
8.381.1 Detailed Description . . . 349
8.381.2 Field Documentation . . . 350
8.381.2.1 messageIndex . . . 350
8.381.2.2 messageTag . . . 350
8.382 qmiWDSDataBearerTechnology Struct Reference . . . 350
8.382.1 Detailed Description . . . 350
8.382.2 Field Documentation . . . 350
8.382.2.1 currentNetwork . . . 350
8.382.2.2 ratMask . . . 350

8.382.2.3 soMask . . . 350
8.383 RFBandInfoElements Struct Reference . . . 350
8.383.1 Detailed Description . . . 350
8.383.2 Field Documentation . . . 350
8.383.2.1 activeBandClass . . . 350
8.383.2.2 activeChannel . . . 350
8.383.2.3 radioInterface . . . 351
8.384 rmTrasnferStaticsReq Struct Reference . . . 351
8.384.1 Detailed Description . . . 351
8.384.2 Field Documentation . . . 351
8.384.2.1 bResetStatistics . . . 351
8.384.2.2 ulMask . . . 351
8.385 sensorData_t Struct Reference . . . 351
8.385.1 Detailed Description . . . 351
8.385.2 Field Documentation . . . 352
8.385.2.1 flags . . . 352
8.385.2.2 sensorDataLen . . . 352
8.385.2.3 timeOfFirstSample . . . 352
8.385.2.4 timeOffset . . . 352
8.385.2.5 xAxis . . . 352
8.385.2.6 yAxis . . . 352
8.385.2.7 zAxis . . . 352
8.386 slot_t Struct Reference . . . 352
8.386.1 Detailed Description . . . 353
8.386.2 Field Documentation . . . 353
8.386.2.1 bICCID . . . 353
8.386.2.2 bICCIDLength . . . 353
8.386.2.3 bLogicalSlot . . . 353
8.386.2.4 uPhyCardStatus . . . 353
8.386.2.5 uPhySlotStatus . . . 353
8.387 slotInf Struct Reference . . . 353
8.387.1 Detailed Description . . . 354
8.387.2 Field Documentation . . . 355
8.387.2.1 AppStatus . . . 355
8.387.2.2 cardState . . . 355
8.387.2.3 errorState . . . 355
8.387.2.4 numApp . . . 355
8.387.2.5 upinRetries . . . 355
8.387.2.6 upinState . . . 355
8.387.2.7 upukRetries . . . 355

8.388 slots_t Struct Reference . . . 355
8.388.1 Field Documentation . . . 355
8.388.1.1 uimSlotStatus . . . 355
8.389 sms_BroadcastConfig Struct Reference . . . 355
8.389.1 Detailed Description . . . 355
8.389.2 Field Documentation . . . 356
8.389.2.1 fromServiceId . . . 356
8.389.2.2 selected . . . 356
8.389.2.3 toServiceId . . . 356
8.390 sms_CDMABroadcastConfig Struct Reference . . . 356
8.390.1 Detailed Description . . . 356
8.390.2 Field Documentation . . . 356
8.390.2.1 language . . . 356
8.390.2.2 selected . . . 356
8.390.2.3 serviceCategory . . . 356
8.391 sms_getIndicationReg Struct Reference . . . 356
8.391.1 Detailed Description . . . 357
8.391.2 Field Documentation . . . 357
8.391.2.1 pRegCallStatInfoEvt . . . 357
8.391.2.2 pRegTransLayerInfoEvt . . . 357
8.391.2.3 pRegTransNWRegInfoEvt . . . 357
8.392 sms_getMsgWaitingInfo Struct Reference . . . 357
8.392.1 Detailed Description . . . 358
8.392.2 Field Documentation . . . 358
8.392.2.1 msgWaitInfo . . . 358
8.392.2.2 numInstances . . . 358
8.393 sms_getTransLayerInfo Struct Reference . . . 358
8.393.1 Detailed Description . . . 358
8.393.2 Field Documentation . . . 358
8.393.2.1 pRegInd . . . 358
8.393.2.2 pTransLayerInfo . . . 358
8.394 sms_getTransNWRegInfo Struct Reference . . . 359
8.394.1 Detailed Description . . . 359
8.394.2 Field Documentation . . . 359
8.394.2.1 pRegStatus . . . 359
8.395 sms_maxStorageSizeReq Struct Reference . . . 359
8.395.1 Detailed Description . . . 359
8.395.2 Field Documentation . . . 360
8.395.2.1 pMessageMode . . . 360
8.395.2.2 storageType . . . 360

8.396 sms_maxStorageSizeResp Struct Reference . . . 360
8.396.1 Detailed Description . . . 360
8.396.2 Field Documentation . . . 360
8.396.2.1 freeSlots . . . 360
8.396.2.2 maxStorageSize . . . 360
8.397 sms_messageWaitingInfoContent Struct Reference . . . 360
8.397.1 Detailed Description . . . 360
8.397.2 Field Documentation . . . 361
8.397.2.1 activeInd . . . 361
8.397.2.2 msgCount . . . 361
8.397.2.3 msgType . . . 361
8.398 sms_msgProtocolResp Struct Reference . . . 361
8.398.1 Detailed Description . . . 361
8.398.2 Field Documentation . . . 361
8.398.2.1 msgProtocol . . . 361
8.399 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference . . . 361
8.399.1 Detailed Description . . . 362
8.399.2 Field Documentation . . . 362
8.399.2.1 activated_ind . . . 362
8.399.2.2 CDMABroadcastConfig . . . 362
8.399.2.3 num_instances . . . 362
8.400 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference . . . 362
8.400.1 Detailed Description . . . 362
8.400.2 Field Documentation . . . 363
8.400.2.1 activated_ind . . . 363
8.400.2.2 broadcastConfig . . . 363
8.400.2.3 num_instances . . . 363
8.401 sms_routeEntry Struct Reference . . . 363
8.401.1 Detailed Description . . . 363
8.401.2 Field Documentation . . . 364
8.401.2.1 messageClass . . . 364
8.401.2.2 messageType . . . 364
8.401.2.3 receiptAction . . . 364
8.401.2.4 routeStorage . . . 364
8.402 sms_sendAsyncsmsParams Struct Reference . . . 364
8.402.1 Detailed Description . . . 365
8.402.2 Field Documentation . . . 366
8.402.2.1 messageFormat . . . 366
8.402.2.2 messageSize . . . 366
8.402.2.3 pFollowOnDC . . . 366

8.402.2.4 pForceOnDC . . . 366
8.402.2.5 pLinktimer . . . 366
8.402.2.6 pMessage . . . 366
8.402.2.7 pRetryMessage . . . 366
8.402.2.8 pRetryMessageId . . . 366
8.402.2.9 pServiceOption . . . 366
8.402.2.10 pSmsOnIms . . . 366
8.402.2.11 pUserData . . . 366
8.403 sms_setIndicationReg Struct Reference . . . 366
8.403.1 Detailed Description . . . 367
8.403.2 Field Documentation . . . 367
8.403.2.1 pRegCallStatInfoEvt . . . 367
8.403.2.2 pRegTransLayerInfoEvt . . . 367
8.403.2.3 pRegTransNWRegInfoEvt . . . 367
8.404 sms_setRoutesReq Struct Reference . . . 367
8.404.1 Detailed Description . . . 368
8.404.2 Field Documentation . . . 368
8.404.2.1 numOfRoutes . . . 368
8.404.2.2 pTransferStatusReport . . . 368
8.404.2.3 routeList . . . 368
8.405 sms_transLayerInfo Struct Reference . . . 368
8.405.1 Detailed Description . . . 368
8.405.2 Field Documentation . . . 368
8.405.2.1 TransCap . . . 368
8.405.2.2 TransType . . . 368
8.406 sMSCAddress Struct Reference . . . 369
8.406.1 Detailed Description . . . 369
8.406.2 Field Documentation . . . 369
8.406.2.1 data . . . 369
8.406.2.2 length . . . 369
8.407 sMSCAddressTlv Struct Reference . . . 369
8.407.1 Detailed Description . . . 369
8.407.2 Field Documentation . . . 369
8.407.2.1 SMSCInfo . . . 369
8.407.2.2 TlvPresent . . . 369
8.408 sMSEtwsMessage Struct Reference . . . 369
8.408.1 Detailed Description . . . 370
8.408.2 Field Documentation . . . 370
8.408.2.1 data . . . 370
8.408.2.2 length . . . 370

8.408.2.3 notificationType . . . 370
8.409 sMSEtwsMessageTlv Struct Reference . . . 370
8.409.1 Detailed Description . . . 370
8.409.2 Field Documentation . . . 370
8.409.2.1 EtwsMessageInfo . . . 370
8.409.2.2 TlvPresent . . . 370
8.410 sMSEtwsPlmn Struct Reference . . . 371
8.410.1 Detailed Description . . . 371
8.410.2 Field Documentation . . . 371
8.410.2.1 mobileCountryCode . . . 371
8.410.2.2 mobileNetworkCode . . . 371
8.411 sMSMessageMode Struct Reference . . . 371
8.411.1 Detailed Description . . . 371
8.411.2 Field Documentation . . . 371
8.411.2.1 messageMode . . . 371
8.412 sMSMTMessage Struct Reference . . . 371
8.412.1 Detailed Description . . . 371
8.412.2 Field Documentation . . . 372
8.412.2.1 messageIndex . . . 372
8.412.2.2 storageType . . . 372
8.413 sMSOnIMS Struct Reference . . . 372
8.413.1 Detailed Description . . . 372
8.413.2 Field Documentation . . . 372
8.413.2.1 smsOnIMS . . . 372
8.414 sMSOnIMSTlv Struct Reference . . . 372
8.414.1 Detailed Description . . . 372
8.414.2 Field Documentation . . . 372
8.414.2.1 IMSInfo . . . 372
8.414.2.2 TlvPresent . . . 372
8.415 sMSTransferRouteMTMessage Struct Reference . . . 373
8.415.1 Detailed Description . . . 373
8.415.2 Field Documentation . . . 373
8.415.2.1 ackIndicator . . . 373
8.415.2.2 data . . . 373
8.415.2.3 format . . . 373
8.415.2.4 length . . . 373
8.415.2.5 transactionID . . . 373
8.416 tdscdmaSigInfoExt Struct Reference . . . 373
8.416.1 Detailed Description . . . 374
8.416.2 Field Documentation . . . 374

8.416.2.1 ecio . . . 374
8.416.2.2 rscp . . . 374
8.416.2.3 rssi . . . 374
8.416.2.4 sinr . . . 374
8.417 tempData_t Struct Reference . . . 374
8.417.1 Detailed Description . . . 374
8.417.2 Field Documentation . . . 375
8.417.2.1 temperature . . . 375
8.417.2.2 temperatureDataLen . . . 375
8.417.2.3 timeOfFirstSample . . . 375
8.417.2.4 timeOffset . . . 375
8.417.2.5 timeSource . . . 375
8.418 transferRouteMessageTlv Struct Reference . . . 375
8.418.1 Detailed Description . . . 375
8.418.2 Field Documentation . . . 375
8.418.2.1 TlvPresent . . . 375
8.418.2.2 TransferRouteMTMessageInfo . . . 375
8.419 transferStatInd Struct Reference . . . 375
8.419.1 Detailed Description . . . 376
8.419.2 Field Documentation . . . 376
8.419.2.1 StatsMask . . . 376
8.419.2.2 StatsPeriod . . . 376
8.420 uim_appStatus Struct Reference . . . 376
8.420.1 Detailed Description . . . 376
8.420.2 Field Documentation . . . 378
8.420.2.1 aidLength . . . 378
8.420.2.2 aidVal . . . 378
8.420.2.3 appState . . . 378
8.420.2.4 appType . . . 378
8.420.2.5 persoFeature . . . 378
8.420.2.6 persoRetries . . . 378
8.420.2.7 persoState . . . 378
8.420.2.8 persoUnblockRetries . . . 379
8.420.2.9 pin1Retries . . . 379
8.420.2.10 pin1State . . . 379
8.420.2.11 pin2Retries . . . 379
8.420.2.12 pin2State . . . 379
8.420.2.13 puk1Retries . . . 379
8.420.2.14 puk2Retries . . . 379
8.420.2.15 univPin . . . 379

8.421 uim_authenticateResult Struct Reference . . . 379
8.421.1 Detailed Description . . . 379
8.421.2 Field Documentation . . . 379
8.421.2.1 content . . . 379
8.421.2.2 contentLen . . . 379
8.422 uim_authenticationData Struct Reference . . . 379
8.422.1 Detailed Description . . . 380
8.422.2 Field Documentation . . . 380
8.422.2.1 context . . . 380
8.422.2.2 data . . . 380
8.422.2.3 dataLen . . . 380
8.423 uim_cardResult Struct Reference . . . 380
8.423.1 Detailed Description . . . 381
8.423.2 Field Documentation . . . 381
8.423.2.1 sw1 . . . 381
8.423.2.2 sw2 . . . 381
8.424 uim_cardStatus Struct Reference . . . 381
8.424.1 Detailed Description . . . 381
8.424.2 Field Documentation . . . 382
8.424.2.1 index1xPri . . . 382
8.424.2.2 index1xSec . . . 382
8.424.2.3 indexGwPri . . . 382
8.424.2.4 indexGwSec . . . 382
8.424.2.5 numSlot . . . 382
8.424.2.6 SlotInfo . . . 382
8.425 uim_changeUIMPIN Struct Reference . . . 382
8.425.1 Detailed Description . . . 382
8.425.2 Field Documentation . . . 383
8.425.2.1 oldPINLen . . . 383
8.425.2.2 oldPINVal . . . 383
8.425.2.3 pinID . . . 383
8.425.2.4 pinLen . . . 383
8.425.2.5 pinVal . . . 383
8.426 uim_depersonalizationInformation Struct Reference . . . 383
8.426.1 Detailed Description . . . 383
8.426.2 Field Documentation . . . 384
8.426.2.1 ckLen . . . 384
8.426.2.2 ckVal . . . 384
8.426.2.3 feature . . . 384
8.426.2.4 operation . . . 384

8.427 uim_encryptedPIN1 Struct Reference . . . 384
8.427.1 Detailed Description . . . 384
8.427.2 Field Documentation . . . 385
8.427.2.1 pin1Len . . . 385
8.427.2.2 pin1Val . . . 385
8.428 uim_fileAttributes Struct Reference . . . 385
8.428.1 Detailed Description . . . 385
8.428.2 Field Documentation . . . 388
8.428.2.1 fileID . . . 388
8.428.2.2 fileSize . . . 388
8.428.2.3 fileType . . . 388
8.428.2.4 rawLen . . . 388
8.428.2.5 rawValue . . . 388
8.428.2.6 recordCount . . . 388
8.428.2.7 recordSize . . . 388
8.428.2.8 secActivate . . . 388
8.428.2.9 secActivateMask . . . 388
8.428.2.10 secDeactivate . . . 388
8.428.2.11 secDeactivateMask . . . 388
8.428.2.12 secIncrease . . . 388
8.428.2.13 secIncreaseMask . . . 388
8.428.2.14 secRead . . . 388
8.428.2.15 secReadMask . . . 388
8.428.2.16 secWrite . . . 388
8.428.2.17 secWriteMask . . . 388
8.429 uim_fileInfo Struct Reference . . . 388
8.429.1 Detailed Description . . . 388
8.429.2 Field Documentation . . . 389
8.429.2.1 fileID . . . 389
8.429.2.2 path . . . 389
8.429.2.3 pathLen . . . 389
8.430 uim_hotSwapStatus Struct Reference . . . 389
8.430.1 Detailed Description . . . 389
8.430.2 Field Documentation . . . 389
8.430.2.1 hotSwap . . . 389
8.430.2.2 hotSwapLength . . . 389
8.431 uim_personalizationStatus Struct Reference . . . 389
8.431.1 Detailed Description . . . 390
8.431.2 Field Documentation . . . 390
8.431.2.1 feature . . . 390

8.431.2.2 numFeatures . . . 390
8.431.2.3 unblockLeft . . . 390
8.431.2.4 verifyLeft . . . 390
8.432uim_readResult Struct Reference . . . 390
8.432.1 Detailed Description . . . 391
8.432.2 Field Documentation . . . 391
8.432.2.1 content . . . 391
8.432.2.2 contentLen . . . 391
8.433uim_readTransparentInfo Struct Reference . . . 391
8.433.1 Detailed Description . . . 391
8.433.2 Field Documentation . . . 391
8.433.2.1 length . . . 391
8.433.2.2 offset . . . 391
8.434uim_refreshevent Struct Reference . . . 391
8.434.1 Detailed Description . . . 392
8.434.2 Field Documentation . . . 393
8.434.2.1 aid . . . 393
8.434.2.2 aidLength . . . 393
8.434.2.3 arrfileInfo . . . 393
8.434.2.4 mode . . . 393
8.434.2.5 numOfFiles . . . 393
8.434.2.6 sessionType . . . 393
8.434.2.7 stage . . . 393
8.435uim_registerRefresh Struct Reference . . . 393
8.435.1 Detailed Description . . . 393
8.435.2 Field Documentation . . . 394
8.435.2.1 arrfileInfo . . . 394
8.435.2.2 numFiles . . . 394
8.435.2.3 registerFlag . . . 394
8.435.2.4 voteForInit . . . 394
8.436uim_remainingRetries Struct Reference . . . 394
8.436.1 Detailed Description . . . 394
8.436.2 Field Documentation . . . 394
8.436.2.1 unblockLeft . . . 394
8.436.2.2 verifyLeft . . . 395
8.437uim_sessionInformation Struct Reference . . . 395
8.437.1 Detailed Description . . . 395
8.437.2 Field Documentation . . . 395
8.437.2.1 aid . . . 395
8.437.2.2 aidLength . . . 395

8.437.2.3 sessionType . . . 395
8.438 uim_setPINProtection Struct Reference . . . 395
8.438.1 Detailed Description . . . 396
8.438.2 Field Documentation . . . 396
8.438.2.1 pinID . . . 396
8.438.2.2 pinLength . . . 396
8.438.2.3 pinOperation . . . 396
8.438.2.4 pinValue . . . 396
8.439 uim_slotInfo Struct Reference . . . 396
8.439.1 Detailed Description . . . 397
8.439.2 Field Documentation . . . 398
8.439.2.1 AppStatus . . . 398
8.439.2.2 cardState . . . 398
8.439.2.3 errorState . . . 398
8.439.2.4 numApp . . . 398
8.439.2.5 upinRetries . . . 398
8.439.2.6 upinState . . . 398
8.439.2.7 upukRetries . . . 398
8.440 uim_UIMSessionInformation Struct Reference . . . 398
8.440.1 Detailed Description . . . 398
8.440.2 Field Documentation . . . 399
8.440.2.1 aid . . . 399
8.440.2.2 aidLength . . . 399
8.440.2.3 sessionType . . . 399
8.441 uim_unblockUIMPIN Struct Reference . . . 399
8.441.1 Detailed Description . . . 399
8.441.2 Field Documentation . . . 399
8.441.2.1 newPINLen . . . 399
8.441.2.2 newPINVal . . . 399
8.441.2.3 pinID . . . 399
8.441.2.4 pukLen . . . 400
8.441.2.5 pukVal . . . 400
8.442 uim_verifyUIMPIN Struct Reference . . . 400
8.442.1 Detailed Description . . . 400
8.442.2 Field Documentation . . . 400
8.442.2.1 pinID . . . 400
8.442.2.2 pinLen . . . 400
8.442.2.3 pinVal . . . 400
8.443 unpack_dms_GetActivationState_t Struct Reference . . . 400
8.443.1 Detailed Description . . . 400

8.443.2 Field Documentation . . . 401
8.443.2.1 state . . . 401
8.444 unpack_dms_GetBandCapability_t Struct Reference . . . 401
8.444.1 Field Documentation . . . 401
8.444.1.1 BandCapability . . . 401
8.444.1.2 Tlvresult . . . 401
8.445 unpack_dms_GetCrashAction_t Struct Reference . . . 401
8.445.1 Field Documentation . . . 401
8.445.1.1 DevCrashState . . . 401
8.445.1.2 Tlvresult . . . 401
8.446 unpack_dms_GetCustFeature_t Struct Reference . . . 401
8.446.1 Field Documentation . . . 402
8.446.1.1 DHCPRelayEnabled . . . 402
8.446.1.2 DisableIMSI . . . 402
8.446.1.3 GpsEnable . . . 402
8.446.1.4 GPSLPM . . . 402
8.446.1.5 GPSSel . . . 402
8.446.1.6 IPFamSupport . . . 402
8.446.1.7 IsVoiceEnabled . . . 402
8.446.1.8 RMAutoConnect . . . 402
8.446.1.9 SMSSupport . . . 402
8.446.1.10 Tlvresult . . . 402
8.447 unpack_dms_GetCustFeaturesV2_t Struct Reference . . . 402
8.447.1 Detailed Description . . . 402
8.447.2 Field Documentation . . . 402
8.447.2.1 GetCustomFeatureV2 . . . 402
8.447.2.2 Tlvresult . . . 402
8.448 unpack_dms_GetDeviceCap_t Struct Reference . . . 403
8.448.1 Field Documentation . . . 403
8.448.1.1 DataServiceCapability . . . 403
8.448.1.2 MaxRXChannelRate . . . 403
8.448.1.3 MaxTXChannelRate . . . 403
8.448.1.4 RadioIfaces . . . 403
8.448.1.5 RadioIfacesSize . . . 403
8.448.1.6 SimCapability . . . 403
8.448.1.7 Tlvresult . . . 403
8.449 unpack_dms_GetDeviceCapabilities_t Struct Reference . . . 403
8.449.1 Detailed Description . . . 403
8.449.2 Field Documentation . . . 404
8.449.2.1 dataServiceCaCapability . . . 404

8.449.2.2 maxRxChannelRate . . . 404
8.449.2.3 maxTxChannelRate . . . 404
8.449.2.4 RadioIfaces . . . 404
8.449.2.5 radioIfacesSize . . . 404
8.449.2.6 simCapability . . . 404
8.450 unpack_dms_GetDeviceHardwareRev_t Struct Reference . . . 404
8.450.1 Field Documentation . . . 404
8.450.1.1 String . . . 404
8.450.1.2 stringSize . . . 404
8.450.1.3 Tlvresult . . . 404
8.451 unpack_dms_GetDeviceMfr_t Struct Reference . . . 404
8.451.1 Field Documentation . . . 404
8.451.1.1 String . . . 404
8.451.1.2 stringSize . . . 404
8.451.1.3 Tlvresult . . . 404
8.452 unpack_dms_GetDeviceSerialNumbers_t Struct Reference . . . 404
8.452.1 Field Documentation . . . 405
8.452.1.1 esnSize . . . 405
8.452.1.2 ESNString . . . 405
8.452.1.3 imeiSize . . . 405
8.452.1.4 IMEIString . . . 405
8.452.1.5 imeiSvnSize . . . 405
8.452.1.6 ImeiSvnString . . . 405
8.452.1.7 meidSize . . . 405
8.452.1.8 MEIDString . . . 405
8.452.1.9 Tlvresult . . . 405
8.453 unpack_dms_GetFirmwareInfo_t Struct Reference . . . 405
8.453.1 Detailed Description . . . 405
8.453.2 Field Documentation . . . 406
8.453.2.1 appversion_str . . . 406
8.453.2.2 bootversion_str . . . 406
8.453.2.3 carrier_str . . . 406
8.453.2.4 cur_carr_name . . . 406
8.453.2.5 cur_carr_rev . . . 406
8.453.2.6 modelid_str . . . 406
8.453.2.7 packageid_str . . . 406
8.453.2.8 priversion_str . . . 406
8.453.2.9 sku_str . . . 406
8.453.2.10 Tlvresult . . . 406
8.454 unpack_dms_GetFirmwareRevision_t Struct Reference . . . 406

8.454.1 Field Documentation . . . 406
8.454.1.1 amssSize . . . 406
8.454.1.2 AMSSString . . . 406
8.454.1.3 PRIString . . . 406
8.454.1.4 Tlvresult . . . 406
8.455 unpack_dms_GetFirmwareRevisions_t Struct Reference . . . 406
8.455.1 Detailed Description . . . 407
8.455.2 Field Documentation . . . 407
8.455.2.1 amssSize . . . 407
8.455.2.2 AMSSString . . . 407
8.455.2.3 bootSize . . . 407
8.455.2.4 BootString . . . 407
8.455.2.5 priSize . . . 407
8.455.2.6 PRIString . . . 407
8.455.2.7 Tlvresult . . . 407
8.456 unpack_dms_GetFSN_t Struct Reference . . . 407
8.456.1 Field Documentation . . . 407
8.456.1.1 String . . . 407
8.456.1.2 Tlvresult . . . 407
8.457 unpack_dms_GetHardwareRevision_t Struct Reference . . . 407
8.457.1 Detailed Description . . . 407
8.457.2 Field Documentation . . . 408
8.457.2.1 hwVer . . . 408
8.458 unpack_dms_GetIMSI_t Struct Reference . . . 408
8.458.1 Field Documentation . . . 408
8.458.1.1 imsi . . . 408
8.458.1.2 Tlvresult . . . 408
8.459 unpack_dms_GetManufacturer_t Struct Reference . . . 408
8.459.1 Detailed Description . . . 408
8.459.2 Field Documentation . . . 408
8.459.2.1 manufacturer . . . 408
8.459.2.2 Tlvresult . . . 408
8.460 unpack_dms_GetModelID_t Struct Reference . . . 408
8.460.1 Detailed Description . . . 409
8.460.2 Field Documentation . . . 409
8.460.2.1 modelid . . . 409
8.460.2.2 Tlvresult . . . 409
8.461 unpack_dms_GetNetworkTime_t Struct Reference . . . 409
8.461.1 Detailed Description . . . 409
8.461.2 Field Documentation . . . 409

8.461.2.1 source . . . 409
8.461.2.2 timestamp . . . 409
8.461.2.3 TlvResult . . . 409
8.462 unpack_dms_GetOfflineReason_t Struct Reference . . . 410
8.462.1 Detailed Description . . . 410
8.462.2 Field Documentation . . . 410
8.462.2.1 pbPlatform . . . 410
8.462.2.2 pReasonMask . . . 410
8.462.2.3 TlvResult . . . 410
8.463 unpack_dms_GetPower_t Struct Reference . . . 410
8.463.1 Detailed Description . . . 410
8.463.2 Field Documentation . . . 411
8.463.2.1 HardwareControlledMode . . . 411
8.463.2.2 OfflineReason . . . 411
8.463.2.3 OperationMode . . . 411
8.463.2.4 TlvResult . . . 411
8.464 unpack_dms_GetPRLVersion_t Struct Reference . . . 411
8.464.1 Field Documentation . . . 411
8.464.1.1 TlvResult . . . 411
8.464.1.2 u16PRLVersion . . . 411
8.464.1.3 u8PRLPreference . . . 411
8.465 unpack_dms_GetSerialNumbers_t Struct Reference . . . 411
8.465.1 Detailed Description . . . 411
8.465.2 Field Documentation . . . 411
8.465.2.1 esn . . . 412
8.465.2.2 imei_no . . . 412
8.465.2.3 imeisv_svn . . . 412
8.465.2.4 meid . . . 412
8.466 unpack_dms_GetUSBComp_t Struct Reference . . . 412
8.466.1 Field Documentation . . . 412
8.466.1.1 NumSupUSBComps . . . 412
8.466.1.2 SupUSBComps . . . 412
8.466.1.3 TlvResult . . . 412
8.466.1.4 USBComp . . . 412
8.467 unpack_dms_GetVoiceNumber_t Struct Reference . . . 412
8.467.1 Field Documentation . . . 412
8.467.1.1 MIN . . . 412
8.467.1.2 minSize . . . 412
8.467.1.3 TlvResult . . . 412
8.467.1.4 VoiceNumber . . . 412

8.467.1.5 voiceNumberSize . . . 412
8.468 unpack_dms_ResetToFactoryDefaults_t Struct Reference . . . 412
8.468.1 Detailed Description . . . 413
8.468.2 Field Documentation . . . 413
8.468.2.1 Tlvresult . . . 413
8.469 unpack_dms_SetActivationStatusCallback_t Struct Reference . . . 413
8.469.1 Detailed Description . . . 413
8.469.2 Field Documentation . . . 413
8.469.2.1 Tlvresult . . . 413
8.470 unpack_dms_SetCrashAction_t Struct Reference . . . 413
8.470.1 Detailed Description . . . 413
8.470.2 Field Documentation . . . 413
8.470.2.1 notused . . . 414
8.471 unpack_dms_SetCustFeature_t Struct Reference . . . 414
8.471.1 Field Documentation . . . 414
8.471.1.1 Tlvresult . . . 414
8.472 unpack_dms_SetCustFeaturesV2_t Struct Reference . . . 414
8.472.1 Detailed Description . . . 414
8.472.2 Field Documentation . . . 414
8.472.2.1 Tlvresult . . . 414
8.473 unpack_dms_SetEventReport_ind_t Struct Reference . . . 414
8.473.1 Detailed Description . . . 414
8.473.2 Field Documentation . . . 415
8.473.2.1 ActivationStatusTlv . . . 415
8.473.2.2 OperatingModeTlv . . . 415
8.473.2.3 Tlvresult . . . 415
8.474 unpack_dms_SetEventReport_t Struct Reference . . . 415
8.474.1 Field Documentation . . . 415
8.474.1.1 Tlvresult . . . 415
8.475 unpack_dms_SetFirmwarePreference_t Struct Reference . . . 415
8.475.1 Field Documentation . . . 415
8.475.1.1 Tlvresult . . . 415
8.476 unpack_dms_SetPower_t Struct Reference . . . 415
8.476.1 Field Documentation . . . 415
8.476.1.1 Tlvresult . . . 415
8.477 unpack_dms_SetUSBComp_t Struct Reference . . . 415
8.477.1 Field Documentation . . . 416
8.477.1.1 Tlvresult . . . 416
8.478 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference . . . 416
8.478.1 Detailed Description . . . 416

8.478.2 Field Documentation . . . 416
8.478.2.1 source . . . 416
8.478.2.2 TlvResult . . . 416
8.478.2.3 type . . . 416
8.479 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference . . . 417
8.479.1 Detailed Description . . . 417
8.479.2 Field Documentation . . . 417
8.479.2.1 source . . . 417
8.479.2.2 TlvResult . . . 417
8.479.2.3 type . . . 417
8.480 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference . . . 417
8.480.1 Detailed Description . . . 418
8.480.2 Field Documentation . . . 418
8.480.2.1 TlvResult . . . 418
8.481 unpack_dms_SLQSGetBandCapability_t Struct Reference . . . 418
8.481.1 Detailed Description . . . 418
8.481.2 Field Documentation . . . 421
8.481.2.1 bandCapability . . . 421
8.481.2.2 is_LteBandCapability_Available . . . 421
8.481.2.3 is_TdsBandCapability_Available . . . 421
8.481.2.4 LteBandCapability . . . 421
8.481.2.5 TdsBandCapability . . . 421
8.482 unpack_dms_SLQSGetERIFile_t Struct Reference . . . 421
8.482.1 Detailed Description . . . 421
8.482.2 Field Documentation . . . 421
8.482.2.1 eriFile . . . 421
8.482.2.2 TlvResult . . . 421
8.483 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference . . . 421
8.483.1 Detailed Description . . . 422
8.483.2 Field Documentation . . . 422
8.483.2.1 TlvResult . . . 422
8.484 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference . . . 422
8.484.1 Detailed Description . . . 422
8.484.2 Field Documentation . . . 422
8.484.2.1 crashInfoParam . . . 422
8.484.2.2 TlvResult . . . 422
8.485 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference . . . 422
8.485.1 Detailed Description . . . 422
8.485.2 Field Documentation . . . 423
8.485.2.1 pGetDyingGaspCfg . . . 423

8.485.2.2 TlvResult . . . 423
8.486 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference . . . 423
8.486.1 Detailed Description . . . 423
8.486.2 Field Documentation . . . 423
8.486.2.1 pGetDyingGaspStatistics . . . 423
8.486.2.2 TlvResult . . . 423
8.487 unpack_dms_SLQSSwiGetFirmwareCurr_t Struct Reference . . . 423
8.487.1 Detailed Description . . . 423
8.487.2 Field Documentation . . . 424
8.487.2.1 carrier . . . 424
8.487.2.2 fwvers . . . 424
8.487.2.3 numEntries . . . 424
8.487.2.4 pCurrImgInfo . . . 424
8.487.2.5 pkgver . . . 424
8.487.2.6 priver . . . 424
8.488 unpack_dms_SLQSSwiGetFwUpdateStatus_t Struct Reference . . . 424
8.488.1 Detailed Description . . . 424
8.488.2 Field Documentation . . . 425
8.488.2.1 imgType . . . 425
8.488.2.2 logString . . . 425
8.488.2.3 refData . . . 425
8.488.2.4 refString . . . 425
8.488.2.5 ResCode . . . 425
8.488.2.6 TlvResult . . . 426
8.489 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference . . . 426
8.489.1 Detailed Description . . . 426
8.489.2 Field Documentation . . . 426
8.489.2.1 manString . . . 426
8.489.2.2 modelString . . . 426
8.489.2.3 plasmaIDString . . . 426
8.489.2.4 swVerString . . . 426
8.489.2.5 TlvResult . . . 426
8.490 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference . . . 426
8.490.1 Detailed Description . . . 427
8.490.2 Field Documentation . . . 427
8.490.2.1 nameString . . . 427
8.490.2.2 TlvResult . . . 427
8.490.2.3 versionString . . . 427
8.491 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference . . . 427
8.491.1 Detailed Description . . . 427

8.491.2 Field Documentation . . . 427
8.491.2.1 meidString . . . 427
8.491.2.2 Tlvresult . . . 427
8.492 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference . . . 427
8.492.1 Detailed Description . . . 428
8.492.2 Field Documentation . . . 428
8.492.2.1 Tlvresult . . . 428
8.493 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference . . . 428
8.493.1 Detailed Description . . . 428
8.493.2 Field Documentation . . . 428
8.493.2.1 Tlvresult . . . 428
8.494 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference . . . 428
8.494.1 Detailed Description . . . 428
8.494.2 Field Documentation . . . 429
8.494.2.1 Tlvresult . . . 429
8.495 unpack_dms_SLQSUIMGetState_t Struct Reference . . . 429
8.495.1 Detailed Description . . . 429
8.495.2 Field Documentation . . . 429
8.495.2.1 state . . . 429
8.495.2.2 Tlvresult . . . 429
8.496 unpack_dms_UIMGetControlKeyStatus_t Struct Reference . . . 429
8.496.1 Detailed Description . . . 429
8.496.2 Field Documentation . . . 430
8.496.2.1 facilityState . . . 430
8.496.2.2 Tlvresult . . . 430
8.496.2.3 unblockRetriesLeft . . . 430
8.496.2.4 verifyRetriesLeft . . . 430
8.497 unpack_dms_UIMGetICCID_t Struct Reference . . . 430
8.497.1 Detailed Description . . . 430
8.497.2 Field Documentation . . . 430
8.497.2.1 String . . . 431
8.497.2.2 stringSize . . . 431
8.497.2.3 Tlvresult . . . 431
8.498 unpack_dms_UIMGetPINStatus_t Struct Reference . . . 431
8.498.1 Detailed Description . . . 431
8.498.2 Field Documentation . . . 432
8.498.2.1 p1Status . . . 432
8.498.2.2 p1UnblockRetriesLeft . . . 432
8.498.2.3 p1VerifyRetriesLeft . . . 432
8.498.2.4 p2Status . . . 432

8.498.2.5 p2UnblockRetriesLeft . . . 432
8.498.2.6 p2VerifyRetriesLeft . . . 432
8.498.2.7 Tlvresult . . . 432
8.499 unpack_dms_UIMSetControlKeyProtection_t Struct Reference . . . 432
8.499.1 Detailed Description . . . 432
8.499.2 Field Documentation . . . 433
8.499.2.1 Tlvresult . . . 433
8.499.2.2 verifyRetriesLeft . . . 433
8.500 unpack_dms_UIMSetPINProtection_t Struct Reference . . . 433
8.500.1 Detailed Description . . . 433
8.500.2 Field Documentation . . . 433
8.500.2.1 Tlvresult . . . 433
8.500.2.2 unblockRetriesLeft . . . 433
8.500.2.3 verifyRetriesLeft . . . 433
8.501 unpack_dms_UIMUnblockControlKey_t Struct Reference . . . 433
8.501.1 Detailed Description . . . 434
8.501.2 Field Documentation . . . 434
8.501.2.1 Tlvresult . . . 434
8.501.2.2 unblockRetriesLeft . . . 434
8.502 unpack_fms_GetImagesPreference_t Struct Reference . . . 434
8.502.1 Detailed Description . . . 434
8.502.2 Field Documentation . . . 434
8.502.2.1 ImageListSize . . . 434
8.502.2.2 pImageList . . . 434
8.502.2.3 Tlvresult . . . 434
8.503 unpack_fms_GetStoredImages_t Struct Reference . . . 435
8.503.1 Detailed Description . . . 435
8.503.2 Field Documentation . . . 435
8.503.2.1 imageList . . . 435
8.503.2.2 imagelistSize . . . 435
8.503.2.3 Tlvresult . . . 435
8.504 unpack_fms_SetImagesPreference_t Struct Reference . . . 435
8.504.1 Detailed Description . . . 435
8.504.2 Field Documentation . . . 436
8.504.2.1 ImageTypes . . . 436
8.504.2.2 ImageTypesSize . . . 436
8.504.2.3 Tlvresult . . . 436
8.505 unpack_loc_BestAvailPos_Ind_t Struct Reference . . . 436
8.505.1 Detailed Description . . . 437
8.505.2 Field Documentation . . . 440

8.505.2.1 pAltitudeWrtEllipsoid . . . 440
8.505.2.2 pAltitudeWrtMeanSeaLevel . . . 440
8.505.2.3 pGpsTime . . . 440
8.505.2.4 pHeading . . . 440
8.505.2.5 pHeadingUnc . . . 440
8.505.2.6 pHorCirConf . . . 440
8.505.2.7 pHorEllpConf . . . 440
8.505.2.8 pHorReliability . . . 441
8.505.2.9 pHorUncCircular . . . 441
8.505.2.10 pHorUncEllipseOrientAzimuth . . . 441
8.505.2.11 pHorUncEllipseSemiMajor . . . 441
8.505.2.12 pHorUncEllipseSemiMinor . . . 441
8.505.2.13 pLatitude . . . 441
8.505.2.14 pLongitude . . . 441
8.505.2.15 pMagneticDeviation . . . 441
8.505.2.16 pPrecisionDilution . . . 441
8.505.2.17 pSensorDataUsage . . . 441
8.505.2.18 pSpeedHorizontal . . . 441
8.505.2.19 pSpeedUnc . . . 441
8.505.2.20 pSpeedVertical . . . 441
8.505.2.21 pSpeedVerticalUnc . . . 441
8.505.2.22 pSvUsedforFix . . . 441
8.505.2.23 pTechnologyMask . . . 441
8.505.2.24 pTimeSrc . . . 441
8.505.2.25 pTimestampUtc . . . 441
8.505.2.26 pTimeUnc . . . 441
8.505.2.27 pVertConfidence . . . 441
8.505.2.28 pVertReliability . . . 441
8.505.2.29 pVertUnc . . . 441
8.505.2.30 pXid . . . 441
8.505.2.31 status . . . 441
8.505.2.32 Tlvresult . . . 441
8.506 unpack_loc_Delete_Assist_Data_t Struct Reference . . . 441
8.506.1 Detailed Description . . . 442
8.506.2 Field Documentation . . . 442
8.506.2.1 Tlvresult . . . 442
8.507 unpack_loc_DeleteAssistData_Ind_t Struct Reference . . . 442
8.507.1 Detailed Description . . . 442
8.507.2 Field Documentation . . . 442
8.507.2.1 status . . . 443

8.507.2.2 TlvResult . . . 443
8.508 unpack_loc_EngineState_Ind_t Struct Reference . . . 443
8.508.1 Detailed Description . . . 443
8.508.2 Field Documentation . . . 443
8.508.2.1 engineState . . . 443
8.508.2.2 TlvResult . . . 443
8.509 unpack_loc_EventRegister_t Struct Reference . . . 443
8.509.1 Detailed Description . . . 443
8.509.2 Field Documentation . . . 443
8.509.2.1 TlvResult . . . 444
8.510 unpack_loc_GnssSvInfo_Ind_t Struct Reference . . . 444
8.510.1 Detailed Description . . . 444
8.510.2 Field Documentation . . . 444
8.510.2.1 altitudeAssumed . . . 444
8.510.2.2 pSatelliteInfo . . . 444
8.510.2.3 TlvResult . . . 444
8.511 unpack_loc_PositionRpt_Ind_t Struct Reference . . . 444
8.511.1 Detailed Description . . . 445
8.511.2 Field Documentation . . . 449
8.511.2.1 pAltitudeAssumed . . . 449
8.511.2.2 pAltitudeWrtEllipsoid . . . 449
8.511.2.3 pAltitudeWrtMeanSeaLevel . . . 449
8.511.2.4 pFixId . . . 449
8.511.2.5 pGpsTime . . . 449
8.511.2.6 pHeading . . . 450
8.511.2.7 pHeadingUnc . . . 450
8.511.2.8 pHorConfidence . . . 450
8.511.2.9 pHorReliability . . . 450
8.511.2.10 pHorUncCircular . . . 450
8.511.2.11 pHorUncEllipseOrientAzimuth . . . 450
8.511.2.12 pHorUncEllipseSemiMajor . . . 450
8.511.2.13 pHorUncEllipseSemiMinor . . . 450
8.511.2.14 pLatitude . . . 450
8.511.2.15 pLeapSeconds . . . 450
8.511.2.16 pLongitude . . . 450
8.511.2.17 pMagneticDeviation . . . 450
8.511.2.18 pPrecisionDilution . . . 450
8.511.2.19 pSensorDataUsage . . . 450
8.511.2.20 pSpeedHorizontal . . . 450
8.511.2.21 pSpeedUnc . . . 450

8.511.2.22 pSpeedVertical . . . 450
8.511.2.23 pSvUsedforFix . . . 450
8.511.2.24 pTechnologyMask . . . 450
8.511.2.25 pTimeSrc . . . 450
8.511.2.26 pTimestampUtc . . . 450
8.511.2.27 pTimeUnc . . . 450
8.511.2.28 pVertConfidence . . . 450
8.511.2.29 pVertReliability . . . 450
8.511.2.30 pVertUnc . . . 450
8.511.2.31 sessionId . . . 450
8.511.2.32 sessionStatus . . . 450
8.511.2.33 Tlvresult . . . 450
8.512 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference . . . 451
8.512.1 Detailed Description . . . 451
8.512.2 Field Documentation . . . 451
8.512.2.1 status . . . 451
8.512.2.2 Tlvresult . . . 451
8.513 unpack_loc_SetExtPowerState_t Struct Reference . . . 451
8.513.1 Detailed Description . . . 451
8.513.2 Field Documentation . . . 452
8.513.2.1 Tlvresult . . . 452
8.514 unpack_loc_SetOperationMode_Ind_t Struct Reference . . . 452
8.514.1 Detailed Description . . . 452
8.514.2 Field Documentation . . . 452
8.514.2.1 status . . . 452
8.514.2.2 Tlvresult . . . 452
8.515 unpack_loc_SetOperationMode_t Struct Reference . . . 452
8.515.1 Detailed Description . . . 452
8.515.2 Field Documentation . . . 453
8.515.2.1 Tlvresult . . . 453
8.516 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference . . . 453
8.516.1 Detailed Description . . . 453
8.516.2 Field Documentation . . . 453
8.516.2.1 Tlvresult . . . 453
8.517 unpack_loc_Start_t Struct Reference . . . 453
8.517.1 Detailed Description . . . 453
8.517.2 Field Documentation . . . 453
8.517.2.1 Tlvresult . . . 453
8.518 unpack_loc_Stop_t Struct Reference . . . 453
8.518.1 Detailed Description . . . 454

8.518.2 Field Documentation . . . 454
8.518.2.1 Tlvresult . . . 454
8.519 unpack_nas_GetCDMANetworkParameters_t Struct Reference . . . 454
8.519.1 Detailed Description . . . 454
8.519.2 Field Documentation . . . 454
8.519.2.1 Application . . . 454
8.519.2.2 Broadcast . . . 455
8.519.2.3 CustomSCP . . . 455
8.519.2.4 ForceRev0 . . . 455
8.519.2.5 Protocol . . . 455
8.519.2.6 RegForeignNID . . . 455
8.519.2.7 RegForeignSID . . . 455
8.519.2.8 RegHomeSID . . . 455
8.519.2.9 Roaming . . . 455
8.519.2.10 SCI . . . 455
8.519.2.11 SCM . . . 455
8.520 unpack_nas_GetHomeNetwork3GPP2_t Struct Reference . . . 455
8.520.1 Detailed Description . . . 455
8.520.2 Field Documentation . . . 456
8.520.2.1 nameSize . . . 456
8.520.2.2 pMCC . . . 456
8.520.2.3 pMNC . . . 456
8.520.2.4 pName . . . 457
8.520.2.5 pNID . . . 457
8.520.2.6 pNw2DescDisp . . . 457
8.520.2.7 pNw2DescEnc . . . 457
8.520.2.8 pNw2DescLen . . . 457
8.520.2.9 pNw2MCC . . . 457
8.520.2.10 pNw2MNC . . . 457
8.520.2.11 pNw2Name . . . 457
8.520.2.12 pSID . . . 457
8.521 unpack_nas_GetHomeNetwork_t Struct Reference . . . 457
8.521.1 Detailed Description . . . 457
8.521.2 Field Documentation . . . 457
8.521.2.1 mcc . . . 457
8.521.2.2 mnc . . . 457
8.521.2.3 name . . . 457
8.521.2.4 nid . . . 457
8.521.2.5 sid . . . 457
8.522 unpack_nas_GetNetworkPreference_t Struct Reference . . . 457

8.522.1 Detailed Description . . . 458
8.522.2 Field Documentation . . . 458
8.522.2.1 ActiveTechPref . . . 458
8.522.2.2 Duration . . . 458
8.522.2.3 PersistentTechPref . . . 458
8.522.2.4 Tlvresult . . . 458
8.523 unpack_nas_GetRFInfo_t Struct Reference . . . 458
8.523.1 Detailed Description . . . 459
8.523.2 Field Documentation . . . 459
8.523.2.1 instancesSize . . . 459
8.523.2.2 RFBandInfoElements . . . 459
8.524 unpack_nas_GetServingNetwork_t Struct Reference . . . 459
8.524.1 Detailed Description . . . 459
8.524.2 Field Documentation . . . 460
8.524.2.1 CSDomain . . . 460
8.524.2.2 DataCaps . . . 460
8.524.2.3 DataCapsLen . . . 460
8.524.2.4 MCC . . . 460
8.524.2.5 MNC . . . 460
8.524.2.6 Name . . . 460
8.524.2.7 nameSize . . . 460
8.524.2.8 PSDomain . . . 460
8.524.2.9 RadioIfaces . . . 460
8.524.2.10 RadioIfacesSize . . . 460
8.524.2.11 RAN . . . 460
8.524.2.12 RegistrationState . . . 460
8.524.2.13 Roaming . . . 460
8.525 unpack_nas_GetServingNetworkCapabilities_t Struct Reference . . . 460
8.525.1 Detailed Description . . . 460
8.525.2 Field Documentation . . . 460
8.525.2.1 DataCaps . . . 460
8.525.2.2 DataCapsLen . . . 460
8.526 unpack_nas_GetSignalStrengths_t Struct Reference . . . 460
8.526.1 Detailed Description . . . 461
8.526.2 Field Documentation . . . 461
8.526.2.1 len . . . 461
8.526.2.2 radio . . . 461
8.526.2.3 rssi . . . 461
8.527 unpack_nas_PerformNetworkScan_t Struct Reference . . . 461
8.527.1 Detailed Description . . . 461

8.527.2 Field Documentation . . . 461
8.527.2.1 p3GppNetworkInfoInstances . . . 461
8.527.2.2 p3GppNetworkInstanceSize . . . 461
8.527.2.3 pPCSInstance . . . 461
8.527.2.4 pPCSInstanceSize . . . 461
8.527.2.5 pRATINstance . . . 461
8.527.2.6 pRATInstanceSize . . . 461
8.527.2.7 pScanResult . . . 461
8.528 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference . . . 462
8.528.1 Detailed Description . . . 462
8.528.2 Field Documentation . . . 462
8.528.2.1 dataCaps . . . 462
8.528.2.2 dataCapsSize . . . 462
8.529 unpack_nas_SetEventReportInd_t Struct Reference . . . 462
8.529.1 Detailed Description . . . 462
8.529.2 Field Documentation . . . 462
8.529.2.1 RFTlv . . . 462
8.529.2.2 RRTlv . . . 462
8.529.2.3 SLQSSSTlv . . . 462
8.529.2.4 SSTlv . . . 462
8.530 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference . . . 462
8.530.1 Detailed Description . . . 463
8.530.2 Field Documentation . . . 463
8.530.2.1 sPhyCaAggPcellInfo . . . 463
8.530.2.2 sPhyCaAggScellDlBw . . . 463
8.530.2.3 sPhyCaAggScellIndex . . . 463
8.530.2.4 sPhyCaAggScellIndType . . . 463
8.530.2.5 sPhyCaAggScellInfo . . . 463
8.531 unpack_nas_SetNetworkPreference_t Struct Reference . . . 463
8.531.1 Detailed Description . . . 463
8.531.2 Field Documentation . . . 464
8.531.2.1 Tlvresult . . . 464
8.532 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference . . . 464
8.532.1 Detailed Description . . . 464
8.532.2 Field Documentation . . . 465
8.532.2.1 roaming . . . 465
8.533 unpack_nas_SetServingSystemCallback_ind_t Struct Reference . . . 465
8.533.1 Detailed Description . . . 465
8.533.2 Field Documentation . . . 465
8.533.2.1 SSInfo . . . 465

8.533.2.2 Tlvresult . . . 465
8.534 unpack_nas_SLQSGetErrorRate_t Struct Reference . . . 465
8.534.1 Detailed Description . . . 465
8.534.2 Field Documentation . . . 466
8.534.2.1 pCDMAFrameErrRate . . . 466
8.534.2.2 pGSMBER . . . 466
8.534.2.3 pHDRPackErrRate . . . 466
8.534.2.4 pWCDMABER . . . 466
8.535 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference . . . 466
8.535.1 Detailed Description . . . 466
8.535.2 Field Documentation . . . 466
8.535.2.1 LTECphyCaInfo . . . 466
8.535.2.2 Tlvresult . . . 466
8.536 unpack_nas_SLQSGetNetworkTime_t Struct Reference . . . 467
8.536.1 Detailed Description . . . 467
8.536.2 Field Documentation . . . 467
8.536.2.1 p3GPP2TimeInfo . . . 467
8.536.2.2 p3GPPTimeInfo . . . 467
8.537 unpack_nas_SLQSGetOperatorNameData_t Struct Reference . . . 467
8.537.1 Detailed Description . . . 467
8.537.2 Field Documentation . . . 468
8.537.2.1 pNITZInformation . . . 468
8.537.2.2 pOperatorNameString . . . 468
8.537.2.3 pOperatorPLMNList . . . 468
8.537.2.4 pPLMNNetworkName . . . 468
8.537.2.5 pSrvcProviderName . . . 468
8.538 unpack_nas_SLQSGetPLMNName_t Struct Reference . . . 468
8.538.1 Field Documentation . . . 468
8.538.1.1 longName . . . 468
8.538.1.2 longNameCI . . . 468
8.538.1.3 longNameEn . . . 468
8.538.1.4 longNameLen . . . 468
8.538.1.5 longNameSB . . . 468
8.538.1.6 shortName . . . 468
8.538.1.7 shortNameCI . . . 469
8.538.1.8 shortNameEn . . . 469
8.538.1.9 shortNameLen . . . 469
8.538.1.10 shortNameSB . . . 469
8.538.1.11 spn . . . 469
8.538.1.12 spnEncoding . . . 469

8.538.1.13 spnLength . . . 469
8.539 unpack_nas_SLQSGetServingSystem_t Struct Reference . . . 469
8.539.1 Detailed Description . . . 469
8.539.2 Field Documentation . . . 470
8.539.2.1 BasestationID . . . 470
8.539.2.2 BasestationLatitude . . . 470
8.539.2.3 BasestationLongitude . . . 470
8.539.2.4 CallBarStatus . . . 470
8.539.2.5 CDMA_P_Rev . . . 470
8.539.2.6 CDMASystemInfoExt . . . 470
8.539.2.7 CellID . . . 470
8.539.2.8 ConcSvcInfo . . . 470
8.539.2.9 CurrentPLMN . . . 470
8.539.2.10 DataSrvCapabilities . . . 470
8.539.2.11 DefaultRoamInd . . . 470
8.539.2.12 DetailedSvcInfo . . . 470
8.539.2.13 DTMInd . . . 470
8.539.2.14 Gpp2TimeZone . . . 471
8.539.2.15 GppNetworkDSTAdjustment . . . 471
8.539.2.16 GppTimeZone . . . 471
8.539.2.17 HdrPersonality . . . 471
8.539.2.18 Lac . . . 471
8.539.2.19 NetworkID . . . 471
8.539.2.20 PRLInd . . . 471
8.539.2.21 RoamIndicatorVal . . . 471
8.539.2.22 RoamingIndicatorList . . . 471
8.539.2.23 ServingSystem . . . 471
8.539.2.24 SystemID . . . 471
8.539.2.25 TrackAreaCode . . . 471
8.540 unpack_nas_SLQSGetSignalStrength_t Struct Reference . . . 471
8.540.1 Detailed Description . . . 471
8.540.2 Field Documentation . . . 472
8.540.2.1 ecioList . . . 472
8.540.2.2 ecioListLen . . . 472
8.540.2.3 errorRateList . . . 472
8.540.2.4 errorRateListLen . . . 472
8.540.2.5 Io . . . 472
8.540.2.6 ltersrp . . . 472
8.540.2.7 ltesnr . . . 472
8.540.2.8 rsrqInfo . . . 472

8.540.2.9 rxSignalStrengthList . . . 472
8.540.2.10 rxSignalStrengthListLen . . . 472
8.540.2.11 signalStrengthReqMask . . . 472
8.540.2.12 sinr . . . 472
8.541 unpack_nas_SLQSGetSysInfo_t Struct Reference . . . 472
8.541.1 Detailed Description . . . 473
8.541.2 Field Documentation . . . 474
8.541.2.1 pAddCDMASysInfo . . . 474
8.541.2.2 pAddGSMSysInfo . . . 474
8.541.2.3 pAddHDRSysInfo . . . 474
8.541.2.4 pAddLTESysInfo . . . 474
8.541.2.5 pAddWCDMASysInfo . . . 474
8.541.2.6 pCDMASrvStatusInfo . . . 474
8.541.2.7 pCDMASysInfo . . . 474
8.541.2.8 pGSMCallBarringSysInfo . . . 474
8.541.2.9 pGSMCipherDomainSysInfo . . . 474
8.541.2.10 pGSMSrvStatusInfo . . . 474
8.541.2.11 pGSMSysInfo . . . 474
8.541.2.12 pHDRSrvStatusInfo . . . 474
8.541.2.13 pHDRSysInfo . . . 474
8.541.2.14 pLTESrvStatusInfo . . . 474
8.541.2.15 pLTESysInfo . . . 475
8.541.2.16 pLTEVoiceSupportSysInfo . . . 475
8.541.2.17 pWCDMACallBarringSysInfo . . . 475
8.541.2.18 pWCDMACipherDomainSysInfo . . . 475
8.541.2.19 pWCDMASrvStatusInfo . . . 475
8.541.2.20 pWCDMASysInfo . . . 475
8.542 unpack_nas_SLQSGetSysSelectionPref_t Struct Reference . . . 475
8.542.1 Detailed Description . . . 475
8.542.2 Field Documentation . . . 478
8.542.2.1 pBandPref . . . 478
8.542.2.2 pEmerMode . . . 478
8.542.2.3 pGWAcqOrderPref . . . 478
8.542.2.4 pLTEBandPref . . . 478
8.542.2.5 pModePref . . . 478
8.542.2.6 pNetSelPref . . . 478
8.542.2.7 pPRLPref . . . 478
8.542.2.8 pRoamPref . . . 478
8.542.2.9 pSrvDomainPref . . . 478
8.543 unpack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference . . . 478

8.543.1 Detailed Description . . . 479
8.543.2 Field Documentation . . . 479
8.543.2.1 pCDMAChannel . . . 479
8.543.2.2 pDirNum . . . 479
8.543.2.3 pHomeSIDNID . . . 479
8.543.2.4 pMinBasedIMSI . . . 479
8.543.2.5 pNAMNameInfo . . . 479
8.543.2.6 pTrueIMSI . . . 479
8.544 unpack_nas_SLQSNasGetCellLocationInfo_t Struct Reference . . . 479
8.544.1 Detailed Description . . . 480
8.544.2 Field Documentation . . . 480
8.544.2.1 pCDMAInfo . . . 480
8.544.2.2 pGERANInfo . . . 480
8.544.2.3 pLTEInfoInterfreq . . . 480
8.544.2.4 pLTEInfoIntrafreq . . . 480
8.544.2.5 pLTEInfoNeighboringGSM . . . 480
8.544.2.6 pLTEInfoNeighboringWCDMA . . . 480
8.544.2.7 pUMTSCellID . . . 480
8.544.2.8 pUMTSInfo . . . 480
8.544.2.9 pWCDMAInfoLTENeighborCell . . . 480
8.545 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference . . . 481
8.545.1 Detailed Description . . . 481
8.545.2 Field Documentation . . . 481
8.545.2.1 pColorCode . . . 481
8.546 unpack_nas_SLQSNasGetSigInfo_t Struct Reference . . . 481
8.546.1 Detailed Description . . . 481
8.546.2 Field Documentation . . . 481
8.546.2.1 CDMASSInfo . . . 481
8.546.2.2 GSMSSInfo . . . 481
8.546.2.3 HDRSSInfo . . . 481
8.546.2.4 LTESSInfo . . . 481
8.546.2.5 WCDMASSInfo . . . 482
8.547 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference . . . 482
8.547.1 Detailed Description . . . 482
8.547.2 Field Documentation . . . 482
8.547.2.1 pRXChain0Info . . . 482
8.547.2.2 pRXChain1Info . . . 482
8.547.2.3 pTXInfo . . . 482
8.548 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference . . . 482
8.548.1 Detailed Description . . . 482

8.548.2 Field Documentation . . . 483
8.548.2.1 pDayItSavAdj . . . 483
8.548.2.2 pRadioInterface . . . 483
8.548.2.3 pTimeZone . . . 483
8.548.2.4 universalTime . . . 483
8.549 unpack_nas_SLQSNasSigInfoCallback_ind_t Struct Reference . . . 483
8.549.1 Detailed Description . . . 483
8.549.2 Field Documentation . . . 484
8.549.2.1 pCDMASigInfo . . . 484
8.549.2.2 pGSMSigInfo . . . 484
8.549.2.3 pHDRSigInfo . . . 484
8.549.2.4 pLTESigInfo . . . 484
8.549.2.5 pRscp . . . 484
8.549.2.6 pTDSCDMASigInfoExt . . . 484
8.549.2.7 pWCDMASigInfo . . . 484
8.550 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference . . . 484
8.550.1 Detailed Description . . . 484
8.550.2 Field Documentation . . . 484
8.550.2.1 pLteEARFCN . . . 484
8.550.2.2 pLtePCI . . . 484
8.550.2.3 pWcdmaUARFCN . . . 484
8.551 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference . . . 485
8.551.1 Detailed Description . . . 485
8.551.2 Field Documentation . . . 485
8.551.2.1 commonInfo . . . 485
8.551.2.2 pLTEInfo . . . 485
8.552 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference . . . 485
8.552.1 Detailed Description . . . 485
8.552.2 Field Documentation . . . 485
8.552.2.1 Info . . . 485
8.552.2.2 Tlvresult . . . 485
8.553 unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference . . . 486
8.553.1 Detailed Description . . . 486
8.553.2 Field Documentation . . . 486
8.553.2.1 t3396_apn . . . 486
8.553.2.2 t3396_plmn_id . . . 486
8.553.2.3 t3396_val . . . 486
8.554 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference . . . 486
8.554.1 Detailed Description . . . 486
8.554.2 Field Documentation . . . 486

8.554.2.1 Info . . . 487
8.554.2.2 TlvResult . . . 487
8.555 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference . . . 487
8.555.1 Detailed Description . . . 487
8.555.2 Field Documentation . . . 487
8.555.2.1 pCurrentPersonality . . . 487
8.555.2.2 pPersonalityListLength . . . 487
8.555.2.3 pProtocolSubtypeElement . . . 487
8.556 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference . . . 487
8.556.1 Detailed Description . . . 487
8.556.2 Field Documentation . . . 488
8.556.2.1 pAppSubType . . . 488
8.556.2.2 pCurrentPrsnlty . . . 488
8.556.2.3 pPersonalityListLength . . . 488
8.556.2.4 pProtoSubTypElmnt . . . 488
8.557 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference . . . 488
8.557.1 Detailed Description . . . 488
8.557.2 Field Documentation . . . 488
8.557.2.1 pDRCParams . . . 489
8.557.2.2 pPilotSetData . . . 489
8.557.2.3 pUATI . . . 489
8.558 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference . . . 489
8.558.1 Detailed Description . . . 489
8.558.2 Field Documentation . . . 489
8.558.2.1 CQIValueCW0 . . . 489
8.558.2.2 CQIValueCW1 . . . 489
8.558.2.3 ValidityCW0 . . . 489
8.558.2.4 ValidityCW1 . . . 489
8.559 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference . . . 489
8.559.1 Detailed Description . . . 490
8.559.2 Field Documentation . . . 490
8.559.2.1 pSccRxInfo . . . 490
8.560 unpack_nas_SLQSSwiNetworkDebug_t Struct Reference . . . 490
8.560.1 Detailed Description . . . 490
8.560.2 Field Documentation . . . 490
8.560.2.1 pDataStatusDetail . . . 490
8.560.2.2 pDeviceConfigDetail . . . 490
8.560.2.3 pNetworkStat1x . . . 490
8.560.2.4 pNetworkStatEVDO . . . 490
8.560.2.5 pObjectVer . . . 491

8.561 unpack_nas_SLQSSysInfoCallback_ind_t Struct Reference . . . 491
8.561.1 Detailed Description . . . 491
8.561.2 Field Documentation . . . 493
8.561.2.1 pAddCDMASysInfo . . . 493
8.561.2.2 pAddGSMSysInfo . . . 493
8.561.2.3 pAddHDRSysInfo . . . 493
8.561.2.4 pAddLTESysInfo . . . 493
8.561.2.5 pAddWCDMASysInfo . . . 493
8.561.2.6 pCDMASrvStatusInfo . . . 493
8.561.2.7 pCDMASysInfo . . . 493
8.561.2.8 pGSMCallBarringSysInfo . . . 493
8.561.2.9 pGSMCipherDomainSysInfo . . . 493
8.561.2.10 pGSMSrvStatusInfo . . . 493
8.561.2.11 pGSMSysInfo . . . 493
8.561.2.12 pHDRSrvStatusInfo . . . 493
8.561.2.13 pHDRSysInfo . . . 493
8.561.2.14 pLTESrvStatusInfo . . . 493
8.561.2.15 pLTESysInfo . . . 493
8.561.2.16 pLTEVoiceSupportSysInfo . . . 493
8.561.2.17 pSysInfoNoChange . . . 493
8.561.2.18 pWCDMACallBarringSysInfo . . . 493
8.561.2.19 pWCDMACipherDomainSysInfo . . . 493
8.561.2.20 pWCDMASrvStatusInfo . . . 493
8.561.2.21 pWCDMASysInfo . . . 493
8.562 unpack_omaDmConfigTlv_t Struct Reference . . . 493
8.562.1 Detailed Description . . . 494
8.562.2 Field Documentation . . . 494
8.562.2.1 alertmsg . . . 494
8.562.2.2 alertmsglength . . . 494
8.562.2.3 state . . . 494
8.562.2.4 userInputReq . . . 494
8.562.2.5 userInputTimeout . . . 494
8.563 unpack_omaDmFotaTlv_t Struct Reference . . . 494
8.563.1 Detailed Description . . . 495
8.563.2 Field Documentation . . . 496
8.563.2.1 description . . . 496
8.563.2.2 descriptionlength . . . 496
8.563.2.3 fwdloadsize . . . 496
8.563.2.4 fwloadComplete . . . 496
8.563.2.5 namelength . . . 496

8.563.2.6 package_name . . . 496
8.563.2.7 sessionType . . . 496
8.563.2.8 severity . . . 496
8.563.2.9 state . . . 496
8.563.2.10 updateCompleteStatus . . . 496
8.563.2.11 userInputReq . . . 496
8.563.2.12 userInputTimeout . . . 496
8.563.2.13 version . . . 496
8.563.2.14 versionlength . . . 496
8.564 unpack_omaDmNotificationsTlv_t Struct Reference . . . 496
8.564.1 Field Documentation . . . 497
8.564.1.1 notification . . . 497
8.564.1.2 sessionStatus . . . 497
8.565 unpack_pds_GetPDSDefaults_t Struct Reference . . . 497
8.565.1 Detailed Description . . . 497
8.565.2 Field Documentation . . . 497
8.565.2.1 pAccuracy . . . 497
8.565.2.2 pInterval . . . 497
8.565.2.3 pOperation . . . 497
8.565.2.4 pTimeout . . . 497
8.566 unpack_pds_GetPDSState_t Struct Reference . . . 497
8.566.1 Detailed Description . . . 498
8.566.2 Field Documentation . . . 498
8.566.2.1 pEnabledStatus . . . 498
8.566.2.2 pTrackingStatus . . . 498
8.567 unpack_pds_GetPortAutomaticTracking_t Struct Reference . . . 498
8.567.1 Detailed Description . . . 498
8.567.2 Field Documentation . . . 498
8.567.2.1 pbAuto . . . 498
8.568 unpack_qmi_t Struct Reference . . . 498
8.568.1 Detailed Description . . . 499
8.568.2 Field Documentation . . . 499
8.568.2.1 msgid . . . 499
8.568.2.2 type . . . 499
8.568.2.3 xid . . . 499
8.569 unpack_qos_dataRate_t Struct Reference . . . 499
8.569.1 Detailed Description . . . 499
8.569.2 Field Documentation . . . 499
8.569.2.1 dataRateMax . . . 499
8.569.2.2 guaranteedRate . . . 499

8.570 unpack_qos_IPv4Addr_t Struct Reference . . . 499
8.570.1 Detailed Description . . . 500
8.570.2 Field Documentation . . . 500
8.570.2.1 addr . . . 500
8.570.2.2 subnetMask . . . 500
8.571 unpack_qos_IPv6Addr_t Struct Reference . . . 500
8.571.1 Detailed Description . . . 500
8.571.2 Field Documentation . . . 500
8.571.2.1 addr . . . 500
8.571.2.2 prefixLen . . . 500
8.572 unpack_qos_IPv6TrafCls_t Struct Reference . . . 500
8.572.1 Detailed Description . . . 500
8.572.2 Field Documentation . . . 501
8.572.2.1 mask . . . 501
8.572.2.2 val . . . 501
8.573 unpack_qos_pktErrRate_t Struct Reference . . . 501
8.573.1 Detailed Description . . . 501
8.573.2 Field Documentation . . . 501
8.573.2.1 exponent . . . 501
8.573.2.2 multiplier . . . 501
8.574 unpack_qos_Port_t Struct Reference . . . 501
8.574.1 Detailed Description . . . 501
8.574.2 Field Documentation . . . 502
8.574.2.1 port . . . 502
8.574.2.2 range . . . 502
8.575 unpack_qos_QosFlowInfo_t Struct Reference . . . 502
8.575.1 Detailed Description . . . 502
8.575.2 Field Documentation . . . 503
8.575.2.1 BearerID . . . 503
8.575.2.2 is_RxQFlowGranted_Available . . . 503
8.575.2.3 is_TxQFlowGranted_Available . . . 503
8.575.2.4 NumRxFilters . . . 503
8.575.2.5 NumTxFilters . . . 503
8.575.2.6 QFlowState . . . 503
8.575.2.7 RxQFilter . . . 503
8.575.2.8 RxQFlowGranted . . . 503
8.575.2.9 TxQFilter . . . 503
8.575.2.10 TxQFlowGranted . . . 503
8.576 unpack_qos_QosFlowInfoState_t Struct Reference . . . 503
8.576.1 Detailed Description . . . 503

8.576.2 Field Documentation . . . 504
8.576.2.1 id . . . 504
8.576.2.2 isNewFlow . . . 504
8.576.2.3 state . . . 504
8.577 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference . . . 504
8.577.1 Detailed Description . . . 504
8.577.2 Field Documentation . . . 504
8.577.2.1 NWQoSStatus . . . 504
8.578 unpack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference . . . 504
8.578.1 Detailed Description . . . 505
8.578.2 Field Documentation . . . 505
8.578.2.1 ambr_dl . . . 505
8.578.2.2 ambr_dl_ext . . . 505
8.578.2.3 ambr_dl_ext2 . . . 505
8.578.2.4 ambr_ul . . . 505
8.578.2.5 ambr_ul_ext . . . 505
8.578.2.6 ambr_ul_ext2 . . . 505
8.578.2.7 apnId . . . 506
8.579 unpack_qos_SLQSQosSwiReadDataStats_t Struct Reference . . . 506
8.579.1 Detailed Description . . . 506
8.579.2 Field Documentation . . . 507
8.579.2.1 apnId . . . 507
8.579.2.2 numQosFlow . . . 507
8.579.2.3 qosFlow . . . 507
8.579.2.4 total_rx_bytes . . . 507
8.579.2.5 total_rx_pkt . . . 507
8.579.2.6 total_tx_bytes . . . 507
8.579.2.7 total_tx_bytes_drp . . . 507
8.579.2.8 total_tx_pkt . . . 507
8.579.2.9 total_tx_pkt_drp . . . 507
8.580 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference . . . 507
8.580.1 Detailed Description . . . 507
8.580.2 Field Documentation . . . 507
8.580.2.1 NumFlows . . . 507
8.580.2.2 QosFlowInfo . . . 507
8.581 unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference . . . 507
8.581.1 Detailed Description . . . 508
8.581.2 Field Documentation . . . 508
8.581.2.1 status . . . 508
8.582 unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference . . . 508

8.582.1 Detailed Description . . . 508
8.582.2 Field Documentation . . . 508
8.582.2.1 event . . . 508
8.583 unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference . . . 508
8.583.1 Detailed Description . . . 508
8.583.2 Field Documentation . . . 509
8.583.2.1 event . . . 509
8.583.2.2 id . . . 509
8.583.2.3 reason . . . 509
8.583.2.4 status . . . 509
8.584 unpack_qos_swiQosFilter_t Struct Reference . . . 510
8.584.1 Detailed Description . . . 510
8.584.2 Field Documentation . . . 512
8.584.2.1 EspSpi . . . 512
8.584.2.2 Id . . . 512
8.584.2.3 index . . . 512
8.584.2.4 IPv4DstAddr . . . 512
8.584.2.5 IPv4SrcAddr . . . 512
8.584.2.6 IPv4Tos . . . 512
8.584.2.7 IPv6DstAddr . . . 512
8.584.2.8 IPv6Label . . . 512
8.584.2.9 IPv6SrcAddr . . . 512
8.584.2.10 IPv6TrafCls . . . 512
8.584.2.11 is_EspSpi_Available . . . 512
8.584.2.12 is_Id_Available . . . 512
8.584.2.13 is_IPv4DstAddr_Available . . . 512
8.584.2.14 is_IPv4SrcAddr_Available . . . 512
8.584.2.15 is_IPv4Tos_Available . . . 512
8.584.2.16 is_IPv6DstAddr_Available . . . 512
8.584.2.17 is_IPv6Label_Available . . . 512
8.584.2.18 is_IPv6SrcAddr_Available . . . 512
8.584.2.19 is_IPv6TrafCls_Available . . . 512
8.584.2.20 is_NxtHdrProto_Available . . . 512
8.584.2.21 is_Precedence_Available . . . 512
8.584.2.22 is_TCPDstPort_Available . . . 512
8.584.2.23 is_TCPSrcPort_Available . . . 513
8.584.2.24 is_TranDstPort_Available . . . 513
8.584.2.25 is_TranSrcPort_Available . . . 513
8.584.2.26 is_UDPDstPort_Available . . . 513
8.584.2.27 is_UDPSrcPort_Available . . . 513

8.584.2.28 NxtHdrProto . . . 513
8.584.2.29 Precedence . . . 513
8.584.2.30 TCPDstPort . . . 513
8.584.2.31 TCPSrcPort . . . 513
8.584.2.32 TranDstPort . . . 513
8.584.2.33 TranSrcPort . . . 513
8.584.2.34 UDPDstPort . . . 513
8.584.2.35 UDPSrcPort . . . 513
8.584.2.36 version . . . 513
8.585 unpack_qos_swiQosFlow_t Struct Reference . . . 513
8.585.1 Detailed Description . . . 514
8.585.2 Field Documentation . . . 516
8.585.2.1 DataRate . . . 516
8.585.2.2 index . . . 516
8.585.2.3 is_DataRate_Available . . . 516
8.585.2.4 is_Jitter_Available . . . 516
8.585.2.5 is_Latency_Available . . . 516
8.585.2.6 is_LteQci_Available . . . 516
8.585.2.7 is_MaxAllowedPktSz_Available . . . 516
8.585.2.8 is_MinPolicedPktSz_Available . . . 516
8.585.2.9 is_PktErrRate_Available . . . 516
8.585.2.10 is_ProfileId3GPP2_Available . . . 516
8.585.2.11 is_TokenBucket_Available . . . 516
8.585.2.12 is_TrafficClass_Available . . . 516
8.585.2.13 is_val_3GPP2Pri_Available . . . 516
8.585.2.14 is_val_3GPPImCn_Available . . . 516
8.585.2.15 is_val_3GPPResResidualBER_Available . . . 516
8.585.2.16 is_val_3GPPSigInd_Available . . . 516
8.585.2.17 is_val_3GPPTraHdlPri_Available . . . 516
8.585.2.18 Jitter . . . 516
8.585.2.19 Latency . . . 517
8.585.2.20 LteQci . . . 517
8.585.2.21 MaxAllowedPktSz . . . 517
8.585.2.22 MinPolicedPktSz . . . 517
8.585.2.23 PktErrRate . . . 517
8.585.2.24 ProfileId3GPP2 . . . 517
8.585.2.25 TokenBucket . . . 517
8.585.2.26 TrafficClass . . . 517
8.585.2.27 val_3GPP2Pri . . . 517
8.585.2.28 val_3GPPImCn . . . 517

8.585.2.29val_3GPPResResidualBER . . . 517
8.585.2.30val_3GPPSigInd . . . 517
8.585.2.31val_3GPPTraHdlPri . . . 517
8.586unpack_qos_tokenBucket_t Struct Reference . . . 517
8.586.1 Detailed Description . . . 517
8.586.2 Field Documentation . . . 517
8.586.2.1 bucketSz . . . 517
8.586.2.2 peakRate . . . 517
8.586.2.3 tokenRate . . . 517
8.587unpack_qos_Tos_t Struct Reference . . . 518
8.587.1 Detailed Description . . . 518
8.587.2 Field Documentation . . . 518
8.587.2.1 mask . . . 518
8.587.2.2 val . . . 518
8.588unpack_QosFlowStat_t Struct Reference . . . 518
8.588.1 Detailed Description . . . 518
8.588.2 Field Documentation . . . 519
8.588.2.1 bearerId . . . 519
8.588.2.2 tx_bytes . . . 519
8.588.2.3 tx_bytes_drp . . . 519
8.588.2.4 tx_pkt . . . 519
8.588.2.5 tx_pkt_drp . . . 519
8.589unpack_rms_GetSMSWake_t Struct Reference . . . 519
8.589.1 Detailed Description . . . 519
8.589.2 Field Documentation . . . 519
8.589.2.1 enabled . . . 519
8.589.2.2 wake_mask . . . 519
8.590unpack_RMTransferStatistics_ind_t Struct Reference . . . 519
8.590.1 Detailed Description . . . 520
8.590.2 Field Documentation . . . 520
8.590.2.1 RxDropConutTlv . . . 520
8.590.2.2 RxOkByteCountTlv . . . 520
8.590.2.3 RxOkConutTlv . . . 520
8.590.2.4 TxDropConutTlv . . . 520
8.590.2.5 TxOkByteCountTlv . . . 520
8.590.2.6 TxOkConutTlv . . . 520
8.591unpack_sar_SLQSGetRfSarState_t Struct Reference . . . 520
8.591.1 Detailed Description . . . 520
8.591.2 Field Documentation . . . 521
8.591.2.1 pSarRFState . . . 521

8.592 unpack_sms_GetSMSCAddress_t Struct Reference . . . 521
8.592.1 Detailed Description . . . 521
8.592.2 Field Documentation . . . 521
8.592.2.1 addressSize . . . 521
8.592.2.2 pSMSCAddress . . . 522
8.592.2.3 pSMSCType . . . 522
8.592.2.4 typeSize . . . 522
8.593 unpack_sms_SaveSMS_t Struct Reference . . . 522
8.593.1 Detailed Description . . . 522
8.593.2 Field Documentation . . . 522
8.593.2.1 pMessageIndex . . . 522
8.594 unpack_sms_SendSMS_t Struct Reference . . . 522
8.594.1 Detailed Description . . . 522
8.594.2 Field Documentation . . . 522
8.594.2.1 messageFailureCode . . . 522
8.594.2.2 messageID . . . 522
8.595 unpack_sms_SetNewSMSCallback_ind_t Struct Reference . . . 522
8.595.1 Detailed Description . . . 523
8.595.2 Field Documentation . . . 523
8.595.2.1 ETWSPLMNTlv . . . 523
8.595.2.2 ETWSTlv . . . 523
8.595.2.3 IMSTlv . . . 523
8.595.2.4 MMTlv . . . 523
8.595.2.5 NewMMTlv . . . 523
8.595.2.6 SMSCTlv . . . 524
8.595.2.7 TRMessageTlv . . . 524
8.596 unpack_sms_SetNewSMSCallback_t Struct Reference . . . 524
8.597 unpack_sms_SLQSDeleteSMS_t Struct Reference . . . 524
8.598 unpack_sms_SLQSGetIndicationRegister_t Struct Reference . . . 524
8.598.1 Detailed Description . . . 524
8.598.2 Field Documentation . . . 524
8.598.2.1 pGetIndicationRegInfo . . . 524
8.599 unpack_sms_SLQSGetMessageWaiting_t Struct Reference . . . 524
8.599.1 Detailed Description . . . 524
8.599.2 Field Documentation . . . 524
8.599.2.1 pGetMsgWaitingInfoResp . . . 524
8.600 unpack_sms_SLQSGetSMS_t Struct Reference . . . 525
8.600.1 Detailed Description . . . 525
8.600.2 Field Documentation . . . 525
8.600.2.1 message . . . 525

8.600.2.2 messageFormat . . . 525
8.600.2.3 messageSize . . . 525
8.600.2.4 messageTag . . . 525
8.601 unpack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference . . . 525
8.601.1 Detailed Description . . . 526
8.601.2 Field Documentation . . . 526
8.601.2.1 pBroadcastConfig . . . 526
8.601.2.2 pCDMABroadcastConfig . . . 526
8.602 unpack_sms_SLQSGetSMSList_t Struct Reference . . . 526
8.602.1 Detailed Description . . . 526
8.602.2 Field Documentation . . . 526
8.602.2.1 messageList . . . 526
8.602.2.2 messageListSize . . . 526
8.603 unpack_sms_SLQSGetTransLayerInfo_t Struct Reference . . . 526
8.603.1 Detailed Description . . . 527
8.603.2 Field Documentation . . . 527
8.603.2.1 pGetTransLayerInfo . . . 527
8.604 unpack_sms_SLQSGetTransNWRegInfo_t Struct Reference . . . 527
8.604.1 Detailed Description . . . 527
8.604.2 Field Documentation . . . 527
8.604.2.1 pGetTransNWRegInfo . . . 527
8.605 unpack_sms_SLQSModifySMSStatus_t Struct Reference . . . 527
8.606 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference . . . 527
8.606.1 Detailed Description . . . 527
8.606.2 Field Documentation . . . 528
8.606.2.1 pMaxStorageSizeResp . . . 528
8.607 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference . . . 528
8.607.1 Detailed Description . . . 528
8.607.2 Field Documentation . . . 528
8.607.2.1 pMessageProtocol . . . 528
8.608 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference . . . 528
8.608.1 Detailed Description . . . 528
8.608.2 Field Documentation . . . 529
8.608.2.1 pSmsStorage . . . 529
8.609 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference . . . 529
8.609.1 Detailed Description . . . 529
8.609.2 Field Documentation . . . 529
8.609.2.1 messageMode . . . 529
8.609.2.2 storageType . . . 529
8.610 unpack_swiavms_SLQSAvMSGetSettings_t Struct Reference . . . 529

8.610.1 Detailed Description . . . 529
8.610.2 Field Documentation . . . 530
8.610.2.1 AutoConnect . . . 530
8.610.2.2 AutoReboot . . . 530
8.610.2.3 FwAutodownload . . . 530
8.610.2.4 FwAutoSDM . . . 530
8.610.2.5 FwAutoUpdate . . . 530
8.610.2.6 OMADMEnabled . . . 530
8.610.2.7 pAPNInfo . . . 530
8.610.2.8 pConnectionRetryTimers . . . 530
8.610.2.9 pNotificationStore . . . 530
8.610.2.10 pPeroidsInfo . . . 530
8.610.2.11 pPollingTimer . . . 530
8.610.2.12 resultcode . . . 530
8.611 unpack_swiavms_SLQSAvMSSendSelection_t Struct Reference . . . 530
8.611.1 Detailed Description . . . 530
8.611.2 Field Documentation . . . 530
8.611.2.1 resultcode . . . 530
8.612 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference . . . 530
8.612.1 Detailed Description . . . 531
8.612.2 Field Documentation . . . 531
8.612.2.1 resultcode . . . 531
8.613 unpack_swiavms_SLQSAvMSSetSettings_t Struct Reference . . . 531
8.613.1 Detailed Description . . . 531
8.613.2 Field Documentation . . . 531
8.613.2.1 resultcode . . . 531
8.614 unpack_swiavms_SLQSAvMSStartSession_t Struct Reference . . . 531
8.614.1 Detailed Description . . . 531
8.614.2 Field Documentation . . . 531
8.614.2.1 resultcode . . . 532
8.614.2.2 sessionResponse . . . 532
8.615 unpack_swiavms_SLQSAvMSStopSession_t Struct Reference . . . 532
8.615.1 Detailed Description . . . 532
8.615.2 Field Documentation . . . 532
8.615.2.1 resultcode . . . 532
8.616 unpack_swiloc_SwiLocGetAutoStart_t Struct Reference . . . 532
8.616.1 Detailed Description . . . 532
8.616.2 Field Documentation . . . 533
8.616.2.1 fix_rate . . . 533
8.616.2.2 fix_rate_reported . . . 533

8.616.2.3 fix_type . . . 533
8.616.2.4 fix_type_reported . . . 533
8.616.2.5 function . . . 533
8.616.2.6 function_reported . . . 533
8.616.2.7 max_dist . . . 533
8.616.2.8 max_dist_reported . . . 534
8.616.2.9 max_time . . . 534
8.616.2.10 max_time_reported . . . 534
8.617 unpack_swioma_SLQSOMADMAlertCallback_ind_t Struct Reference . . . 534
8.617.1 Detailed Description . . . 534
8.617.2 Field Documentation . . . 534
8.617.2.1 eventType . . . 534
8.617.2.2 SessionInfoConfig . . . 534
8.617.2.3 SessionInfoFota . . . 534
8.617.2.4 SessionInfoNotification . . . 534
8.618 unpack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference . . . 534
8.618.1 Detailed Description . . . 535
8.618.2 Field Documentation . . . 536
8.618.2.1 Date . . . 536
8.618.2.2 DateLength . . . 536
8.618.2.3 PkgDescLength . . . 536
8.618.2.4 PkgDescription . . . 537
8.618.2.5 PkgName . . . 537
8.618.2.6 PkgNameLength . . . 537
8.618.2.7 RetryCount . . . 537
8.618.2.8 SessionState . . . 537
8.618.2.9 SessionType . . . 537
8.618.2.10 Severity . . . 537
8.618.2.11 Source . . . 537
8.618.2.12 SourceLength . . . 537
8.618.2.13 Status . . . 537
8.618.2.14 Time . . . 537
8.618.2.15 TimeLength . . . 537
8.618.2.16 UpdateCompleteStatus . . . 537
8.619 unpack_swioma_SLQSOMADMGetSettings_t Struct Reference . . . 537
8.619.1 Detailed Description . . . 537
8.619.2 Field Documentation . . . 538
8.619.2.1 Autosdm . . . 538
8.619.2.2 FOTAdownload . . . 538
8.619.2.3 FOTAUpdate . . . 538

8.619.2.4 FwAutoCheck . . . 538
8.619.2.5 OMADMEnabled . . . 538
8.620 unpack_swioma_SLQSOMADMStartSession_t Struct Reference . . . 538
8.620.1 Detailed Description . . . 539
8.620.2 Field Documentation . . . 539
8.620.2.1 FwAvailability . . . 539
8.621 unpack_uim_ChangePin_t Struct Reference . . . 539
8.621.1 Detailed Description . . . 539
8.621.2 Field Documentation . . . 539
8.621.2.1 pEncryptedPIN1 . . . 539
8.621.2.2 pIndicationToken . . . 540
8.621.2.3 pRemainingRetries . . . 540
8.621.2.4 Tlvresult . . . 540
8.622 unpack_uim_GetCardStatus_t Struct Reference . . . 540
8.622.1 Detailed Description . . . 540
8.622.2 Field Documentation . . . 540
8.622.2.1 pCardStatus . . . 540
8.622.2.2 pHotSwapStatus . . . 540
8.622.2.3 Tlvresult . . . 540
8.623 unpack_uim_ReadTransparent_t Struct Reference . . . 540
8.623.1 Detailed Description . . . 540
8.623.2 Field Documentation . . . 541
8.623.2.1 pCardResult . . . 541
8.623.2.2 pEncryptedData . . . 541
8.623.2.3 pIndicationToken . . . 541
8.623.2.4 pReadResult . . . 541
8.623.2.5 Tlvresult . . . 541
8.624 unpack_uim_SetPinProtection_t Struct Reference . . . 541
8.624.1 Detailed Description . . . 541
8.624.2 Field Documentation . . . 542
8.624.2.1 pEncryptedPIN1 . . . 542
8.624.2.2 pIndicationToken . . . 542
8.624.2.3 pRemainingRetries . . . 542
8.624.2.4 Tlvresult . . . 542
8.625 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference . . . 542
8.625.1 Detailed Description . . . 542
8.625.2 Field Documentation . . . 542
8.625.2.1 bNumberOfPhySlots . . . 542
8.625.2.2 slotsstatusChange . . . 542
8.626 unpack_uim_SLQSUIMAuthenticate_t Struct Reference . . . 542

8.626.1 Detailed Description . . . 543
8.626.2 Field Documentation . . . 543
8.626.2.1 pAuthenticateResult . . . 543
8.626.2.2 pCardResult . . . 543
8.626.2.3 pIndicationToken . . . 543
8.627 unpack_uim_SLQSUIMDepersonalization_t Struct Reference . . . 543
8.627.1 Detailed Description . . . 543
8.627.2 Field Documentation . . . 543
8.627.2.1 pRemainingRetries . . . 544
8.628 unpack_uim_SLQSUIMEventRegister_t Struct Reference . . . 544
8.628.1 Detailed Description . . . 544
8.628.2 Field Documentation . . . 544
8.628.2.1 eventMask . . . 544
8.629 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference . . . 544
8.629.1 Detailed Description . . . 544
8.629.2 Field Documentation . . . 545
8.629.2.1 pAutoSelection . . . 545
8.629.2.2 pHaltSubscription . . . 545
8.629.2.3 pPersonalizationStatus . . . 545
8.630 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference . . . 545
8.630.1 Detailed Description . . . 545
8.630.2 Field Documentation . . . 545
8.630.2.1 pCardResult . . . 545
8.630.2.2 pFileAttributes . . . 545
8.630.2.3 pIndicationToken . . . 545
8.631 unpack_uim_SLQSUIMGetSlotsStatus_t Struct Reference . . . 545
8.631.1 Detailed Description . . . 546
8.631.2 Field Documentation . . . 546
8.631.2.1 pNumberOfPhySlot . . . 546
8.631.2.2 pUimSlotsStatus . . . 546
8.632 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference . . . 546
8.632.1 Detailed Description . . . 546
8.632.2 Field Documentation . . . 546
8.632.2.1 pRefreshEvent . . . 546
8.633 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference . . . 546
8.633.1 Detailed Description . . . 546
8.633.2 Field Documentation . . . 547
8.633.2.1 pCardStatus . . . 547
8.634 unpack_uim_UnblockPin_t Struct Reference . . . 547
8.634.1 Detailed Description . . . 547

8.634.2 Field Documentation . . . 547
8.634.2.1 pEncryptedPIN1 . . . 547
8.634.2.2 pIndicationToken . . . 547
8.634.2.3 pRemainingRetries . . . 547
8.634.2.4 Tlvresult . . . 547
8.635 unpack_uim_VerifyPin_t Struct Reference . . . 547
8.635.1 Detailed Description . . . 547
8.635.2 Field Documentation . . . 548
8.635.2.1 pEncryptedPIN1 . . . 548
8.635.2.2 pIndicationToken . . . 548
8.635.2.3 pRemainingRetries . . . 548
8.635.2.4 Tlvresult . . . 548
8.636 unpack_voice_SLQSOriginateUSSD_t Struct Reference . . . 548
8.636.1 Field Documentation . . . 548
8.636.1.1 pAlphaIDInfo . . . 548
8.636.1.2 pCallId . . . 548
8.636.1.3 pCcResultType . . . 548
8.636.1.4 pCCSuppsType . . . 548
8.636.1.5 pfailureCause . . . 548
8.636.1.6 pUSSDInfo . . . 548
8.637 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference . . . 549
8.637.1 Detailed Description . . . 549
8.637.2 Field Documentation . . . 549
8.637.2.1 pCallId . . . 549
8.638 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference . . . 549
8.638.1 Detailed Description . . . 549
8.638.2 Field Documentation . . . 549
8.638.2.1 pCallID . . . 549
8.639 unpack_voice_SLQSVoiceDialCall_t Struct Reference . . . 549
8.639.1 Detailed Description . . . 550
8.639.2 Field Documentation . . . 550
8.639.2.1 pAlphaIDInfo . . . 550
8.639.2.2 pCallID . . . 550
8.639.2.3 pCCResultType . . . 550
8.639.2.4 pCCSUPSType . . . 550
8.640 unpack_voice_SLQSVoiceEndCall_t Struct Reference . . . 550
8.640.1 Detailed Description . . . 550
8.640.2 Field Documentation . . . 550
8.640.2.1 pCallId . . . 550
8.641 unpack_voice_SLQSVoiceGetAllCallInfo_t Struct Reference . . . 551

8.641.1 Detailed Description . . . 551
8.641.2 Field Documentation . . . 553
8.641.2.1 pArrAlertingPattern . . . 553
8.641.2.2 pArrAlertingType . . . 553
8.641.2.3 pArrAlphaID . . . 553
8.641.2.4 pArrCalledPartyNum . . . 553
8.641.2.5 pArrCallEndReason . . . 553
8.641.2.6 pArrCallInfo . . . 553
8.641.2.7 pArrConnectPartyNum . . . 553
8.641.2.8 pArrDiagInfo . . . 553
8.641.2.9 pArrRedirPartyNum . . . 553
8.641.2.10 pArrRemotePartyName . . . 553
8.641.2.11 pArrRemotePartyNum . . . 553
8.641.2.12 pArrSvcOption . . . 553
8.641.2.13 pArrUUSInfo . . . 553
8.641.2.14 pOTASPStatus . . . 553
8.641.2.15 pVoicePrivacy . . . 553
8.642 unpack_voice_SLQSVoiceGetCallBarring_t Struct Reference . . . 553
8.642.1 Detailed Description . . . 553
8.642.2 Field Documentation . . . 554
8.642.2.1 pAlphaIDInfo . . . 554
8.642.2.2 pCallID . . . 554
8.642.2.3 pCCResType . . . 554
8.642.2.4 pCCSUPSType . . . 554
8.642.2.5 pFailCause . . . 554
8.642.2.6 pSvcClass . . . 554
8.643 unpack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference . . . 555
8.643.1 Detailed Description . . . 555
8.643.2 Field Documentation . . . 556
8.643.2.1 pAlphaIDInfo . . . 556
8.643.2.2 pCallID . . . 556
8.643.2.3 pCCResType . . . 556
8.643.2.4 pCCSUPSType . . . 556
8.643.2.5 pFailCause . . . 556
8.643.2.6 pGetCallFWExtInfo . . . 556
8.643.2.7 pGetCallFWInfo . . . 556
8.644 unpack_voice_SLQSVoiceGetCallInfo_t Struct Reference . . . 556
8.644.1 Detailed Description . . . 556
8.644.2 Field Documentation . . . 558
8.644.2.1 pAlertingPattern . . . 558

8.644.2.2 pAlertType . . . 558
8.644.2.3 pAlphaIDInfo . . . 558
8.644.2.4 pCallInfo . . . 558
8.644.2.5 pConnectNumInfo . . . 558
8.644.2.6 pDiagInfo . . . 558
8.644.2.7 pOTASPStatus . . . 558
8.644.2.8 pRemotePartyName . . . 558
8.644.2.9 pRemotePartyNum . . . 558
8.644.2.10 pSrvOpt . . . 558
8.644.2.11 pUUSInfo . . . 558
8.644.2.12 pVoicePrivacy . . . 558
8.645 unpack_voice_SLQSVoiceGetCallWaiting_t Struct Reference . . . 558
8.645.1 Detailed Description . . . 559
8.645.2 Field Documentation . . . 559
8.645.2.1 pAlphaIDInfo . . . 560
8.645.2.2 pCallID . . . 560
8.645.2.3 pCCResType . . . 560
8.645.2.4 pCCSUPSType . . . 560
8.645.2.5 pFailCause . . . 560
8.645.2.6 pSvcClass . . . 560
8.646 unpack_voice_SLQSVoiceGetCLIP_t Struct Reference . . . 560
8.646.1 Detailed Description . . . 560
8.646.2 Field Documentation . . . 561
8.646.2.1 pAlphaIDInfo . . . 561
8.646.2.2 pCallID . . . 561
8.646.2.3 pCCResType . . . 561
8.646.2.4 pCCSUPSType . . . 561
8.646.2.5 pCLIPResp . . . 561
8.646.2.6 pFailCause . . . 561
8.647 unpack_voice_SLQSVoiceGetCLIR_t Struct Reference . . . 561
8.647.1 Detailed Description . . . 561
8.647.2 Field Documentation . . . 562
8.647.2.1 pAlphaIDInfo . . . 562
8.647.2.2 pCallID . . . 562
8.647.2.3 pCCResType . . . 562
8.647.2.4 pCCSUPSType . . . 562
8.647.2.5 pCLIRResp . . . 562
8.647.2.6 pFailCause . . . 562
8.648 unpack_voice_SLQSVoiceGetCNAP_t Struct Reference . . . 562
8.648.1 Detailed Description . . . 562

8.648.2 Field Documentation . . . 563
8.648.2.1 pAlphaIDInfo . . . 563
8.648.2.2 pCallID . . . 563
8.648.2.3 pCCResType . . . 563
8.648.2.4 pCCSUPSType . . . 563
8.648.2.5 pCNAPResp . . . 563
8.648.2.6 pFailCause . . . 563
8.649 unpack_voice_SLQSVoiceGetCOLP_t Struct Reference . . . 563
8.649.1 Detailed Description . . . 564
8.649.2 Field Documentation . . . 564
8.649.2.1 pAlphaIDInfo . . . 564
8.649.2.2 pCallID . . . 565
8.649.2.3 pCCResType . . . 565
8.649.2.4 pCCSUPSType . . . 565
8.649.2.5 pCOLPResp . . . 565
8.649.2.6 pFailCause . . . 565
8.650 unpack_voice_SLQSVoiceGetCOLR_t Struct Reference . . . 565
8.650.1 Detailed Description . . . 565
8.650.2 Field Documentation . . . 566
8.650.2.1 pAlphaIDInfo . . . 566
8.650.2.2 pCallID . . . 566
8.650.2.3 pCCResType . . . 566
8.650.2.4 pCCSUPSType . . . 566
8.650.2.5 pCOLRResp . . . 566
8.650.2.6 pFailCause . . . 566
8.651 unpack_voice_SLQSVoiceGetConfig_t Struct Reference . . . 566
8.651.1 Detailed Description . . . 566
8.651.2 Field Documentation . . . 567
8.651.2.1 pAirTimerCnt . . . 568
8.651.2.2 pAutoAnswerStat . . . 568
8.651.2.3 pCurAMRConfig . . . 568
8.651.2.4 pCurPrefVoiceSO . . . 568
8.651.2.5 pCurrTTYMode . . . 568
8.651.2.6 pCurVoiceDomainPref . . . 568
8.651.2.7 pCurVoicePrivacyPref . . . 568
8.651.2.8 pRoamTimerCnt . . . 568
8.652 unpack_voice_SLQSVoiceManageCalls_t Struct Reference . . . 568
8.652.1 Detailed Description . . . 568
8.652.2 Field Documentation . . . 568
8.652.2.1 pFailCause . . . 568

8.653 unpack_voice_SLQSVoiceSendFlash_t Struct Reference . . . 568
8.653.1 Detailed Description . . . 568
8.653.2 Field Documentation . . . 569
8.653.2.1 pCallID . . . 569
8.654 unpack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference . . . 569
8.654.1 Detailed Description . . . 569
8.654.2 Field Documentation . . . 569
8.654.2.1 pAlphaIDInfo . . . 570
8.654.2.2 pCallID . . . 570
8.654.2.3 pCCResType . . . 570
8.654.2.4 pCCSUPSType . . . 570
8.654.2.5 pFailCause . . . 570
8.655 unpack_voice_SLQSVoiceSetConfig_t Struct Reference . . . 570
8.655.1 Detailed Description . . . 570
8.655.2 Field Documentation . . . 571
8.655.2.1 pAirTimerStatus . . . 571
8.655.2.2 pAutoAnsStatus . . . 571
8.655.2.3 pPrefVoiceSOStatus . . . 571
8.655.2.4 pRoamTimerStatus . . . 571
8.655.2.5 pTTYConfigStatus . . . 571
8.655.2.6 pVoiceDomainPrefStatus . . . 571
8.656 unpack_voice_SLQSVoiceSetSUPSService_t Struct Reference . . . 571
8.656.1 Detailed Description . . . 572
8.656.2 Field Documentation . . . 572
8.656.2.1 pAlphaIDInfo . . . 572
8.656.2.2 pCallID . . . 572
8.656.2.3 pCCResultType . . . 572
8.656.2.4 pCCSUPSType . . . 572
8.656.2.5 pFailCause . . . 572
8.657 unpack_voice_SLQSVoiceStartContDTMF_t Struct Reference . . . 572
8.657.1 Detailed Description . . . 573
8.657.2 Field Documentation . . . 573
8.657.2.1 pCallID . . . 573
8.658 unpack_voice_SLQSVoiceStopContDTMF_t Struct Reference . . . 573
8.658.1 Detailed Description . . . 573
8.658.2 Field Documentation . . . 573
8.658.2.1 callID . . . 573
8.659 unpack_wds_DHCPv4ClientLease_ind_t Struct Reference . . . 573
8.659.1 Field Documentation . . . 574
8.659.1.1 DHCPv4LeaseOptTlv . . . 574

8.659.1.2 DHCPv4LeaseStateTlv . . . 574
8.659.1.3 IPv4AddrTlv . . . 574
8.659.1.4 ProfileIdTlv . . . 574
8.660 unpack_wds_GetAutoconnect_t Struct Reference . . . 574
8.660.1 Detailed Description . . . 574
8.660.2 Field Documentation . . . 574
8.660.2.1 psetting . . . 574
8.661 unpack_wds_GetByteTotals_t Struct Reference . . . 574
8.661.1 Detailed Description . . . 574
8.661.2 Field Documentation . . . 574
8.661.2.1 pRXTotalBytes . . . 575
8.661.2.2 pTXTotalBytes . . . 575
8.662 unpack_wds_GetConnectionRate_t Struct Reference . . . 575
8.662.1 Detailed Description . . . 575
8.662.2 Field Documentation . . . 575
8.662.2.1 currentChannelRXRate . . . 575
8.662.2.2 currentChannelTXRate . . . 575
8.662.2.3 maxChannelRXRate . . . 575
8.662.2.4 maxChannelTXRate . . . 575
8.663 unpack_wds_GetDataBearerTechnology_t Struct Reference . . . 575
8.663.1 Detailed Description . . . 575
8.663.2 Field Documentation . . . 576
8.663.2.1 pDataBearer . . . 576
8.664 unpack_wds_GetDefaultProfile_t Struct Reference . . . 576
8.664.1 Detailed Description . . . 576
8.664.2 Field Documentation . . . 577
8.664.2.1 apnname . . . 577
8.664.2.2 apnsize . . . 577
8.664.2.3 auth . . . 577
8.664.2.4 ipaddr . . . 577
8.664.2.5 ipaddrv6 . . . 577
8.664.2.6 name . . . 577
8.664.2.7 namesize . . . 577
8.664.2.8 pdptype . . . 577
8.664.2.9 pridns . . . 577
8.664.2.10 pridnsv6 . . . 577
8.664.2.11 secdns . . . 577
8.664.2.12 secdnsv6 . . . 577
8.664.2.13 username . . . 577
8.664.2.14 usersize . . . 577

8.665 unpack_wds_GetDefaultProfileNum_t Struct Reference . . . 577
8.665.1 Detailed Description . . . 577
8.665.2 Field Documentation . . . 578
8.665.2.1 index . . . 578
8.666 unpack_wds_GetDormancyState_t Struct Reference . . . 578
8.666.1 Detailed Description . . . 578
8.666.2 Field Documentation . . . 578
8.666.2.1 dormancyState . . . 578
8.667 unpack_wds_GetLastMobileIPError_t Struct Reference . . . 578
8.667.1 Detailed Description . . . 578
8.667.2 Field Documentation . . . 578
8.667.2.1 error . . . 578
8.668 unpack_wds_GetMobileIP_t Struct Reference . . . 578
8.668.1 Detailed Description . . . 578
8.668.2 Field Documentation . . . 579
8.668.2.1 mipMode . . . 579
8.669 unpack_wds_GetMobileIPProfile_t Struct Reference . . . 579
8.669.1 Detailed Description . . . 579
8.669.2 Field Documentation . . . 579
8.669.2.1 AAASPI . . . 579
8.669.2.2 AAAState . . . 579
8.669.2.3 address . . . 579
8.669.2.4 enabled . . . 579
8.669.2.5 HASPI . . . 579
8.669.2.6 HAState . . . 579
8.669.2.7 NAI . . . 579
8.669.2.8 naiSize . . . 579
8.669.2.9 primaryHA . . . 580
8.669.2.10 revTunneling . . . 580
8.669.2.11 secondaryHA . . . 580
8.670 unpack_wds_GetPacketStatistics_t Struct Reference . . . 580
8.670.1 Detailed Description . . . 580
8.670.2 Field Documentation . . . 581
8.670.2.1 pRXDroppedCount . . . 581
8.670.2.2 pRXOkBytesCount . . . 581
8.670.2.3 pRXOKBytesLastCall . . . 581
8.670.2.4 pRXPacketErrors . . . 581
8.670.2.5 pRXPacketOverflows . . . 581
8.670.2.6 pRXPacketSuccesses . . . 581
8.670.2.7 pTXDroppedCount . . . 581

8.670.2.8 pTXOkBytesCount . . . 581
8.670.2.9 pTXOKBytesLastCall . . . 581
8.670.2.10 pTXPacketErrors . . . 581
8.670.2.11 pTXPacketOverflows . . . 581
8.670.2.12 pTXPacketSuccesses . . . 581
8.671 unpack_wds_GetPacketStatus_t Struct Reference . . . 581
8.671.1 Detailed Description . . . 582
8.671.2 Field Documentation . . . 582
8.671.2.1 rXDroppedCount . . . 582
8.671.2.2 rXOkBytesCount . . . 582
8.671.2.3 rXOKBytesLastCall . . . 582
8.671.2.4 rXPacketErrors . . . 582
8.671.2.5 rXPacketOverflows . . . 582
8.671.2.6 rXPacketSuccesses . . . 582
8.671.2.7 tXDroppedCount . . . 582
8.671.2.8 tXOkBytesCount . . . 582
8.671.2.9 tXOKBytesLastCall . . . 582
8.671.2.10 tXPacketErrors . . . 582
8.671.2.11 tXPacketOverflows . . . 582
8.671.2.12 tXPacketSuccesses . . . 582
8.672 unpack_wds_GetSessionDuration_t Struct Reference . . . 582
8.672.1 Detailed Description . . . 583
8.672.2 Field Documentation . . . 583
8.672.2.1 callDuration . . . 583
8.673 unpack_wds_GetSessionState_t Struct Reference . . . 583
8.673.1 Detailed Description . . . 583
8.673.2 Field Documentation . . . 583
8.673.2.1 connectionStatus . . . 583
8.674 unpack_wds_RMSetTransferStatistics_t Struct Reference . . . 583
8.675 unpack_wds_SetMobileIPProfile_t Struct Reference . . . 583
8.676 unpack_wds_SLQSCreateProfile_t Struct Reference . . . 583
8.676.1 Detailed Description . . . 583
8.676.2 Field Documentation . . . 583
8.676.2.1 pCreateProfileOut . . . 583
8.676.2.2 pProfileID . . . 584
8.676.2.3 Tlvresult . . . 584
8.677 unpack_wds_SLQSDeleteProfile_t Struct Reference . . . 584
8.677.1 Detailed Description . . . 584
8.677.2 Field Documentation . . . 584
8.677.2.1 extendedErrorCode . . . 584

8.678 unpack_wds_SLQSGet3GPPConfigItem_t Struct Reference . . . 584
8.678.1 Detailed Description . . . 584
8.678.2 Field Documentation . . . 585
8.678.2.1 _3gppRelease . . . 585
8.678.2.2 defaultPDNEnabled . . . 585
8.678.2.3 LTEAttachProfile . . . 585
8.678.2.4 LTEAttachProfileList . . . 585
8.678.2.5 LTEAttachProfileListLen . . . 585
8.678.2.6 profileList . . . 585
8.679 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference . . . 585
8.679.1 Detailed Description . . . 585
8.679.2 Field Documentation . . . 586
8.679.2.1 currNetworkInfo . . . 586
8.679.2.2 networkInfoLen . . . 586
8.679.2.3 prefNetwork . . . 586
8.680 unpack_wds_SLQSGetCurrentChannelRate_t Struct Reference . . . 586
8.680.1 Detailed Description . . . 586
8.680.2 Field Documentation . . . 587
8.680.2.1 current_channel_rx_rate . . . 587
8.680.2.2 current_channel_tx_rate . . . 587
8.680.2.3 max_channel_rx_rate . . . 587
8.680.2.4 max_channel_tx_rate . . . 587
8.681 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference . . . 587
8.681.1 Detailed Description . . . 587
8.681.2 Field Documentation . . . 587
8.681.2.1 curDataBearerTechnology . . . 587
8.681.2.2 dataBearerMask . . . 587
8.681.2.3 lastCallDataBearerTechnology . . . 587
8.682 unpack_wds_SLQSGetDUNCallInfo_t Struct Reference . . . 587
8.682.1 Detailed Description . . . 588
8.682.2 Field Documentation . . . 588
8.682.2.1 callEndReason . . . 588
8.682.2.2 channelRate . . . 588
8.682.2.3 connectionStatus . . . 588
8.682.2.4 dataBearerTech . . . 588
8.682.2.5 dormancyStatus . . . 588
8.682.2.6 lastCallDataBearerTech . . . 588
8.682.2.7 lastCallRXOKBytesCnt . . . 588
8.682.2.8 lastCallTXOKBytesCnt . . . 588
8.682.2.9 mdmCallDurationActive . . . 588

8.682.2.10 rxOKBytesCount . . . 588
8.682.2.11 txOKBytesCount . . . 588
8.683 unpack_wds_SLQSGetProfileSettings_t Struct Reference . . . 588
8.683.1 Field Documentation . . . 588
8.683.1.1 pProfileSettings . . . 589
8.683.1.2 ProfileType . . . 589
8.683.1.3 Tlvresult . . . 589
8.684 unpack_wds_SLQSGetRuntimeSettings_t Struct Reference . . . 589
8.684.1 Detailed Description . . . 589
8.684.2 Field Documentation . . . 590
8.684.2.1 APNName . . . 590
8.684.2.2 Authentication . . . 590
8.684.2.3 DomainList . . . 590
8.684.2.4 GPRSGrantedQoS . . . 590
8.684.2.5 GWAddressV4 . . . 590
8.684.2.6 IMCNflag . . . 590
8.684.2.7 IPFamilyPreference . . . 590
8.684.2.8 IPv4 . . . 590
8.684.2.9 IPV6AddrInfo . . . 590
8.684.2.10 IPV6GWAddrInfo . . . 590
8.684.2.11 Mtu . . . 590
8.684.2.12 PCSCFAddrPCO . . . 590
8.684.2.13 PCSCFFQDNAddrList . . . 590
8.684.2.14 PDPType . . . 590
8.684.2.15 PrimaryDNSV4 . . . 590
8.684.2.16 PrimaryDNSV6 . . . 590
8.684.2.17 ProfileID . . . 590
8.684.2.18 ProfileName . . . 590
8.684.2.19 SecondaryDNSV4 . . . 591
8.684.2.20 SecondaryDNSV6 . . . 591
8.684.2.21 ServerAddrList . . . 591
8.684.2.22 SubnetMaskV4 . . . 591
8.684.2.23 Technology . . . 591
8.684.2.24 UMTSGrantedQoS . . . 591
8.684.2.25 Username . . . 591
8.685 unpack_wds_SLQSModifyProfile_t Struct Reference . . . 591
8.685.1 Detailed Description . . . 591
8.685.2 Field Documentation . . . 591
8.685.2.1 pExtErrorCode . . . 591
8.686 unpack_wds_SLQSSetIPFamilyPreference_t Struct Reference . . . 591

8.686.1 Detailed Description . . . 591
8.686.2 Field Documentation . . . 591
8.686.2.1 TlvResult . . . 591
8.687 unpack_wds_SLQSSetPacketSrvStatusCallback_t Struct Reference . . . 591
8.687.1 Detailed Description . . . 592
8.687.2 Field Documentation . . . 592
8.687.2.1 bearerID . . . 592
8.687.2.2 conn_status . . . 592
8.687.2.3 ipFamily . . . 592
8.687.2.4 reconfigReqd . . . 592
8.687.2.5 sessionEndReason . . . 592
8.687.2.6 techName . . . 592
8.687.2.7 verboseSessnEndReason . . . 592
8.687.2.8 verboseSessnEndReasonType . . . 592
8.688 unpack_wds_SLQSSetWdsEventCallback_ind_t Struct Reference . . . 592
8.688.1 Detailed Description . . . 593
8.688.2 Field Documentation . . . 593
8.688.2.1 currDBTechAvail . . . 593
8.688.2.2 currNWInfo . . . 593
8.688.2.3 dataSysStatAvail . . . 593
8.688.2.4 dBTechAvail . . . 594
8.688.2.5 dBtechExtAvail . . . 594
8.688.2.6 dBTechExtRatValue . . . 594
8.688.2.7 dBTechExtSoMask . . . 594
8.688.2.8 dBTechnology . . . 594
8.688.2.9 dBtechnologyExt . . . 594
8.688.2.10 dormancyStatAvail . . . 594
8.688.2.11 dormancyStatus . . . 594
8.688.2.12 mipstatAvail . . . 594
8.688.2.13 mipStatus . . . 594
8.688.2.14 netInfoLen . . . 594
8.688.2.15 prefNetwork . . . 594
8.688.2.16 ratMask . . . 594
8.688.2.17 rx_bytes . . . 594
8.688.2.18 rx_pkts . . . 594
8.688.2.19 soMask . . . 594
8.688.2.20 tx_bytes . . . 594
8.688.2.21 tx_pkts . . . 594
8.688.2.22 xferStatAvail . . . 594
8.689 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference . . . 594

8.689.1 Detailed Description . . . 594
8.689.2 Field Documentation . . . 594
8.689.2.1 pHwConfig . . . 595
8.689.2.2 pRequestOptionList . . . 595
8.690 unpack_wds_SLQSSGetLoopback_t Struct Reference . . . 595
8.690.1 Detailed Description . . . 595
8.690.2 Field Documentation . . . 595
8.690.2.1 ByteLoopbackMode . . . 595
8.690.2.2 ByteLoopbackMultiplier . . . 595
8.691 unpack_wds_SLQSStartDataSession_t Struct Reference . . . 595
8.691.1 Detailed Description . . . 595
8.691.2 Field Documentation . . . 596
8.691.2.1 pFailureReason . . . 596
8.691.2.2 psid . . . 596
8.691.2.3 pVerboseFailReasonType . . . 596
8.691.2.4 pVerboseFailureReason . . . 596
8.692 unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference . . . 596
8.692.1 Detailed Description . . . 596
8.692.2 Field Documentation . . . 597
8.692.2.1 apnName . . . 597
8.692.2.2 bearerId . . . 597
8.692.2.3 contextId . . . 597
8.692.2.4 ipv4Address . . . 597
8.692.2.5 ipv4GWAddress . . . 597
8.692.2.6 ipv6Address . . . 597
8.692.2.7 ipv6GWAddress . . . 597
8.692.2.8 prDNSIPv4Address . . . 597
8.692.2.9 prDNSIPv6Address . . . 597
8.692.2.10 prPCSCFIPv4Address . . . 597
8.692.2.11 prPCSCFIPv6Address . . . 597
8.692.2.12 seDNSIPv4Address . . . 597
8.692.2.13 seDNSIPv6Address . . . 597
8.692.2.14 sePCSCFIPv4Address . . . 597
8.692.2.15 sePCSCFIPv6Address . . . 597
8.693 UnPackGetProfileSettingOut Struct Reference . . . 597
8.693.1 Field Documentation . . . 598
8.693.1.1 curProfile . . . 598
8.693.1.2 pExtErrCode . . . 598
8.694 unpackWdsProfileParam Union Reference . . . 598
8.694.1 Field Documentation . . . 598

8.694.1.1 SlqsProfile3GPP . . . 598
8.694.1.2 SlqsProfile3GPP2 . . . 598
8.695 voice_airTimer Struct Reference . . . 598
8.695.1 Detailed Description . . . 598
8.695.2 Field Documentation . . . 598
8.695.2.1 airTimerValue . . . 598
8.695.2.2 namID . . . 598
8.696 voice_allCallsAlphaIDInfo Struct Reference . . . 598
8.696.1 Detailed Description . . . 599
8.696.2 Field Documentation . . . 599
8.696.2.1 AlphaIDInfo . . . 599
8.696.2.2 callID . . . 599
8.697 voice_allCallsDiagInfo Struct Reference . . . 599
8.697.1 Detailed Description . . . 599
8.697.2 Field Documentation . . . 599
8.697.2.1 callID . . . 599
8.697.2.2 DiagInfo . . . 599
8.698 voice_allCallsUUSInfo Struct Reference . . . 599
8.698.1 Detailed Description . . . 600
8.698.2 Field Documentation . . . 600
8.698.2.1 callID . . . 600
8.698.2.2 uusInfo . . . 600
8.699 voice_alphaIDInfo Struct Reference . . . 600
8.699.1 Detailed Description . . . 600
8.699.2 Field Documentation . . . 601
8.699.2.1 alphaDcs . . . 601
8.699.2.2 alphaLen . . . 601
8.699.2.3 alphaText . . . 601
8.700 voice_arrAlertingPattern Struct Reference . . . 601
8.700.1 Detailed Description . . . 601
8.700.2 Field Documentation . . . 601
8.700.2.1 alertingPattern . . . 601
8.700.2.2 callID . . . 601
8.700.2.3 numInstances . . . 601
8.701 voice_arrAlertingType Struct Reference . . . 601
8.701.1 Detailed Description . . . 602
8.701.2 Field Documentation . . . 602
8.701.2.1 AlertingType . . . 602
8.701.2.2 callID . . . 602
8.701.2.3 numInstances . . . 602

8.702voice_arrAlphaID Struct Reference . . . 602
8.702.1 Detailed Description . . . 602
8.702.2 Field Documentation . . . 603
8.702.2.1 allCallsAlphaIDInfoArr . . . 603
8.702.2.2 numInstances . . . 603
8.703voice_arrCalledPartyNum Struct Reference . . . 603
8.703.1 Detailed Description . . . 603
8.703.2 Field Documentation . . . 603
8.703.2.1 CalledPartyNum . . . 603
8.703.2.2 numInstances . . . 603
8.704voice_arrCallEndReason Struct Reference . . . 603
8.704.1 Detailed Description . . . 603
8.704.2 Field Documentation . . . 604
8.704.2.1 callEndReason . . . 604
8.704.2.2 callID . . . 604
8.704.2.3 numInstances . . . 604
8.705voice_arrCallInfo Struct Reference . . . 604
8.705.1 Detailed Description . . . 604
8.705.2 Field Documentation . . . 604
8.705.2.1 getAllCallInfo . . . 604
8.705.2.2 numInstances . . . 604
8.706voice_arrConnectPartyNum Struct Reference . . . 604
8.706.1 Detailed Description . . . 605
8.706.2 Field Documentation . . . 605
8.706.2.1 ConnectedPartyNum . . . 605
8.706.2.2 numInstances . . . 605
8.707voice_arrDiagInfo Struct Reference . . . 605
8.707.1 Detailed Description . . . 605
8.707.2 Field Documentation . . . 605
8.707.2.1 DiagInfo . . . 605
8.707.2.2 numInstances . . . 605
8.708voice_arrRedirPartyNum Struct Reference . . . 605
8.708.1 Detailed Description . . . 606
8.708.2 Field Documentation . . . 606
8.708.2.1 numInstances . . . 606
8.708.2.2 RedirPartyNum . . . 606
8.709voice_arrRemotePartyName Struct Reference . . . 606
8.709.1 Detailed Description . . . 606
8.709.2 Field Documentation . . . 606
8.709.2.1 GetAllCallRmtPtyName . . . 606

8.709.2.2 numInstances . . . 607
8.710 voice_arrRemotePartyNum Struct Reference . . . 607
8.710.1 Detailed Description . . . 607
8.710.2 Field Documentation . . . 607
8.710.2.1 numInstances . . . 607
8.710.2.2 RmtPtyNum . . . 607
8.711 voice_arrSvcOption Struct Reference . . . 607
8.711.1 Detailed Description . . . 607
8.711.2 Field Documentation . . . 608
8.711.2.1 callID . . . 608
8.711.2.2 numInstances . . . 608
8.711.2.3 srvOption . . . 608
8.712 voice_arrUUSInfo Struct Reference . . . 608
8.712.1 Detailed Description . . . 608
8.712.2 Field Documentation . . . 608
8.712.2.1 AllCallsUUSInfo . . . 608
8.712.2.2 numInstances . . . 608
8.713 voice_burstDTMFInfo Struct Reference . . . 608
8.713.1 Detailed Description . . . 609
8.713.2 Field Documentation . . . 609
8.713.2.1 digitCnt . . . 609
8.713.2.2 pCallID . . . 609
8.713.2.3 pDigitBuff . . . 609
8.714 voice_calledPartySubAdd Struct Reference . . . 609
8.714.1 Detailed Description . . . 609
8.714.2 Field Documentation . . . 610
8.714.2.1 extBit . . . 610
8.714.2.2 oddEvenInd . . . 610
8.714.2.3 subAddr . . . 610
8.714.2.4 subAddrLen . . . 610
8.714.2.5 subAddrType . . . 610
8.715 voice_callFwdTypeAndPlan Struct Reference . . . 610
8.715.1 Detailed Description . . . 610
8.715.2 Field Documentation . . . 611
8.715.2.1 numberPlan . . . 611
8.715.2.2 numberType . . . 611
8.716 voice_callFWExtInfo Struct Reference . . . 611
8.716.1 Detailed Description . . . 611
8.716.2 Field Documentation . . . 612
8.716.2.1 noReplyTimer . . . 612

8.716.2.2 number . . . 612
8.716.2.3 numLen . . . 613
8.716.2.4 numPlan . . . 613
8.716.2.5 numType . . . 613
8.716.2.6 PI . . . 613
8.716.2.7 SI . . . 613
8.716.2.8 SvcClass . . . 613
8.716.2.9 SvcStatus . . . 613
8.717 voice_callFWInfo Struct Reference . . . 613
8.717.1 Detailed Description . . . 613
8.717.2 Field Documentation . . . 613
8.717.2.1 noReplyTimer . . . 613
8.717.2.2 number . . . 614
8.717.2.3 numLen . . . 614
8.717.2.4 SvcClass . . . 614
8.717.2.5 SvcStatus . . . 614
8.718 voice_callInfo Struct Reference . . . 614
8.718.1 Detailed Description . . . 614
8.718.2 Field Documentation . . . 615
8.718.2.1 callID . . . 615
8.718.2.2 callState . . . 615
8.718.2.3 callType . . . 615
8.718.2.4 direction . . . 615
8.718.2.5 mode . . . 615
8.719 voice_ccSUPSType Struct Reference . . . 615
8.719.1 Detailed Description . . . 615
8.719.2 Field Documentation . . . 616
8.719.2.1 reason . . . 616
8.719.2.2 svcType . . . 616
8.720 voice_CLIPResp Struct Reference . . . 616
8.720.1 Detailed Description . . . 616
8.720.2 Field Documentation . . . 617
8.720.2.1 ActiveStatus . . . 617
8.720.2.2 ProvisionStatus . . . 617
8.721 voice_CLIRResp Struct Reference . . . 617
8.721.1 Detailed Description . . . 617
8.721.2 Field Documentation . . . 617
8.721.2.1 ActiveStatus . . . 617
8.721.2.2 ProvisionStatus . . . 617
8.722 voice_CNAPResp Struct Reference . . . 617

8.722.1 Detailed Description . . . 618
8.722.2 Field Documentation . . . 618
8.722.2.1 ActiveStatus . . . 618
8.722.2.2 ProvisionStatus . . . 618
8.723 voice_COLPResp Struct Reference . . . 618
8.723.1 Detailed Description . . . 618
8.723.2 Field Documentation . . . 619
8.723.2.1 ActiveStatus . . . 619
8.723.2.2 ProvisionStatus . . . 619
8.724 voice_COLRResp Struct Reference . . . 619
8.724.1 Detailed Description . . . 619
8.724.2 Field Documentation . . . 619
8.724.2.1 ActiveStatus . . . 619
8.724.2.2 ProvisionStatus . . . 619
8.725 voice_connectNumInfo Struct Reference . . . 620
8.725.1 Detailed Description . . . 620
8.725.2 Field Documentation . . . 621
8.725.2.1 callerID . . . 621
8.725.2.2 callerIDLen . . . 621
8.725.2.3 numPlan . . . 621
8.725.2.4 numPresInd . . . 621
8.725.2.5 numType . . . 621
8.725.2.6 screeningInd . . . 621
8.726 voice_CUGInfo Struct Reference . . . 621
8.726.1 Detailed Description . . . 621
8.726.2 Field Documentation . . . 622
8.726.2.1 CUGIndex . . . 622
8.726.2.2 SuppOA . . . 622
8.726.2.3 SuppPrefCUG . . . 622
8.727 voice_curAMRConfig Struct Reference . . . 622
8.727.1 Detailed Description . . . 622
8.727.2 Field Documentation . . . 622
8.727.2.1 gsmAmrStat . . . 623
8.727.2.2 wcdmaAmrStat . . . 623
8.728 voice_diagInfo Struct Reference . . . 623
8.728.1 Detailed Description . . . 623
8.728.2 Field Documentation . . . 623
8.728.2.1 diagInfoLen . . . 623
8.728.2.2 diagnosticInfo . . . 623
8.729 voice_DTMFLengths Struct Reference . . . 623

8.729.1 Detailed Description . . . 623
8.729.2 Field Documentation . . . 624
8.729.2.1 DTMFInterdigitInterval . . . 624
8.729.2.2 DTMFPulseWidth . . . 624
8.730 voice_getAllCallInformation Struct Reference . . . 624
8.730.1 Detailed Description . . . 624
8.730.2 Field Documentation . . . 625
8.730.2.1 ALS . . . 625
8.730.2.2 CallInfo . . . 625
8.730.2.3 isEmpty . . . 625
8.731 voice_getAllCallRmtPtyName Struct Reference . . . 625
8.731.1 Detailed Description . . . 625
8.731.2 Field Documentation . . . 625
8.731.2.1 callID . . . 625
8.731.2.2 RemotePartyName . . . 625
8.732 voice_getAllCallRmtPtyNum Struct Reference . . . 625
8.732.1 Detailed Description . . . 625
8.732.2 Field Documentation . . . 626
8.732.2.1 callID . . . 626
8.732.2.2 RemotePartyNum . . . 626
8.733 voice_getCallFWExtInfo Struct Reference . . . 626
8.733.1 Detailed Description . . . 626
8.733.2 Field Documentation . . . 626
8.733.2.1 CallFWExtInfo . . . 626
8.733.2.2 numInstances . . . 626
8.734 voice_getCallFWInfo Struct Reference . . . 626
8.734.1 Detailed Description . . . 626
8.734.2 Field Documentation . . . 627
8.734.2.1 CallFWInfo . . . 627
8.734.2.2 numInstances . . . 627
8.735 voice_peerNumberInfo Struct Reference . . . 627
8.735.1 Detailed Description . . . 627
8.735.2 Field Documentation . . . 628
8.735.2.1 callID . . . 628
8.735.2.2 number . . . 628
8.735.2.3 numLen . . . 628
8.735.2.4 numPI . . . 628
8.735.2.5 numPlan . . . 628
8.735.2.6 numSI . . . 628
8.735.2.7 numType . . . 628

8.736 voice_prefVoiceSO Struct Reference . . . 628
8.736.1 Detailed Description . . . 629
8.736.2 Field Documentation . . . 630
8.736.2.1 evrcCapability . . . 630
8.736.2.2 homeOrigVoiceSO . . . 630
8.736.2.3 homePageVoiceSO . . . 630
8.736.2.4 namID . . . 630
8.736.2.5 roamOrigVoiceSO . . . 630
8.737 voice_remotePartyName Struct Reference . . . 630
8.737.1 Detailed Description . . . 631
8.737.2 Field Documentation . . . 631
8.737.2.1 callerName . . . 631
8.737.2.2 codingScheme . . . 631
8.737.2.3 nameLen . . . 631
8.737.2.4 namePI . . . 631
8.738 voice_remotePartyNum Struct Reference . . . 631
8.738.1 Detailed Description . . . 632
8.738.2 Field Documentation . . . 632
8.738.2.1 numLen . . . 632
8.738.2.2 presentationInd . . . 632
8.738.2.3 remPartyNumber . . . 632
8.739 voice_roamTimer Struct Reference . . . 632
8.739.1 Detailed Description . . . 632
8.739.2 Field Documentation . . . 633
8.739.2.1 namID . . . 633
8.739.2.2 roamTimerValue . . . 633
8.740 voice_USSInfo Struct Reference . . . 633
8.740.1 Detailed Description . . . 633
8.740.2 Field Documentation . . . 633
8.740.2.1 ussData . . . 633
8.740.2.2 ussDCS . . . 633
8.740.2.3 ussLen . . . 633
8.741 voice_UUSInfo Struct Reference . . . 633
8.741.1 Detailed Description . . . 634
8.741.2 Field Documentation . . . 634
8.741.2.1 UUSData . . . 634
8.741.2.2 UUSDatalen . . . 634
8.741.2.3 UUSDcs . . . 634
8.741.2.4 UUSType . . . 634
8.742 wds_currNetworkInfo Struct Reference . . . 635

8.742.1 Detailed Description . . . 635
8.742.2 Field Documentation . . . 636
8.742.2.1 NetworkType . . . 636
8.742.2.2 RATMask . . . 636
8.742.2.3 SOMask . . . 636
8.743wds_DataULongLongTlv Struct Reference . . . 636
8.743.1 Field Documentation . . . 636
8.743.1.1 TlvPresent . . . 636
8.743.1.2 ullData . . . 636
8.744wds_DataULongTlv Struct Reference . . . 636
8.744.1 Field Documentation . . . 637
8.744.1.1 TlvPresent . . . 637
8.744.1.2 ulData . . . 637
8.745wds_DHCPLeaseOptTlv Struct Reference . . . 637
8.745.1 Field Documentation . . . 637
8.745.1.1 numOpt . . . 637
8.745.1.2 optList . . . 637
8.745.1.3 optListData . . . 637
8.745.1.4 TlvPresent . . . 637
8.746wds_DHCPLeaseStateTlv Struct Reference . . . 637
8.746.1 Field Documentation . . . 637
8.746.1.1 leaseState . . . 637
8.746.1.2 TlvPresent . . . 637
8.747wds_DHCPOpt Struct Reference . . . 637
8.747.1 Field Documentation . . . 638
8.747.1.1 optCode . . . 638
8.747.1.2 optValLen . . . 638
8.747.1.3 pOptVal . . . 638
8.748wds_DHCPProfileIdTlv Struct Reference . . . 638
8.748.1 Field Documentation . . . 638
8.748.1.1 profileId . . . 638
8.748.1.2 profileType . . . 638
8.748.1.3 TlvPresent . . . 638
8.749wds_DHCPv4HWConfig Struct Reference . . . 638
8.749.1 Detailed Description . . . 638
8.749.2 Field Documentation . . . 638
8.749.2.1 chaddr . . . 638
8.749.2.2 chaddrLen . . . 639
8.749.2.3 hwType . . . 639
8.750wds_DHCPv4Option Struct Reference . . . 639

8.750.1 Detailed Description . . . 639
8.750.2 Field Documentation . . . 639
8.750.2.1 optCode . . . 639
8.750.2.2 optVal . . . 639
8.750.2.3 optValLen . . . 639
8.751 wds_DHCPv4OptionList Struct Reference . . . 639
8.751.1 Detailed Description . . . 639
8.751.2 Field Documentation . . . 640
8.751.2.1 numOpt . . . 640
8.751.2.2 pOptList . . . 640
8.752 wds_DHCPv4ProfileId Struct Reference . . . 640
8.752.1 Detailed Description . . . 640
8.752.2 Field Documentation . . . 640
8.752.2.1 profileId . . . 640
8.752.2.2 profileType . . . 640
8.753 wds_Domain Struct Reference . . . 640
8.753.1 Detailed Description . . . 640
8.753.2 Field Documentation . . . 641
8.753.2.1 domainLen . . . 641
8.753.2.2 domainName . . . 641
8.754 wds_DomainNameList Struct Reference . . . 641
8.754.1 Detailed Description . . . 641
8.754.2 Field Documentation . . . 641
8.754.2.1 domain . . . 641
8.754.2.2 numInstances . . . 641
8.755 wds_GPRSQoS Struct Reference . . . 641
8.755.1 Detailed Description . . . 642
8.755.2 Field Documentation . . . 642
8.755.2.1 delayClass . . . 642
8.755.2.2 meanThroughputClass . . . 642
8.755.2.3 peakThroughputClass . . . 642
8.755.2.4 precedenceClass . . . 642
8.755.2.5 reliabilityClass . . . 642
8.756 wds_IPv4AdTlv Struct Reference . . . 642
8.756.1 Field Documentation . . . 642
8.756.1.1 IPv4Addr . . . 642
8.756.1.2 TlvPresent . . . 642
8.757 wds_IPV6AddressInfo Struct Reference . . . 642
8.757.1 Detailed Description . . . 643
8.757.2 Field Documentation . . . 643

8.757.2.1 IPAddressV6 . . . 643
8.757.2.2 IPV6PrefixLen . . . 643
8.758 wds_IPV6GWAddressInfo Struct Reference . . . 643
8.758.1 Detailed Description . . . 643
8.758.2 Field Documentation . . . 643
8.758.2.1 gwAddressV6 . . . 643
8.758.2.2 gwV6PrefixLen . . . 643
8.759 wds_PCSCFFQDNAddress Struct Reference . . . 643
8.759.1 Detailed Description . . . 644
8.759.2 Field Documentation . . . 644
8.759.2.1 fqdnAddr . . . 644
8.759.2.2 fqdnLen . . . 644
8.760 wds_PCSCFFQDNAddressList Struct Reference . . . 644
8.760.1 Detailed Description . . . 644
8.760.2 Field Documentation . . . 644
8.760.2.1 numInstances . . . 644
8.760.2.2 pcsfFQDNAddress . . . 644
8.761 wds_PCSCFIPv4ServerAddressList Struct Reference . . . 644
8.761.1 Detailed Description . . . 645
8.761.2 Field Documentation . . . 645
8.761.2.1 numInstances . . . 645
8.761.2.2 pscsfIPv4Addr . . . 645
8.762 wds_ProfileIdentifier Struct Reference . . . 645
8.762.1 Detailed Description . . . 645
8.762.2 Field Documentation . . . 645
8.762.2.1 profileIndex . . . 645
8.762.2.2 profileType . . . 645
8.763 wds_profileInfo Union Reference . . . 645
8.763.1 Detailed Description . . . 646
8.763.2 Field Documentation . . . 646
8.763.2.1 SlqsProfile3GPP . . . 646
8.763.2.2 SlqsProfile3GPP2 . . . 646
8.764 wds_TrStatInd Struct Reference . . . 646
8.764.1 Detailed Description . . . 646
8.764.2 Field Documentation . . . 646
8.764.2.1 statsMask . . . 646
8.764.2.2 statsPeriod . . . 646
8.765 wds_UMTSMinQoS Struct Reference . . . 646
8.765.1 Detailed Description . . . 647
8.765.2 Field Documentation . . . 648

8.765.2.1 deliveryErrSDU . . . 648
8.765.2.2 grntDownlinkBitrate . . . 649
8.765.2.3 grntUplinkBitrate . . . 649
8.765.2.4 maxDownlinkBitrate . . . 649
8.765.2.5 maxSDUSize . . . 649
8.765.2.6 maxUplinkBitrate . . . 649
8.765.2.7 qosDeliveryOrder . . . 649
8.765.2.8 resBerRatio . . . 649
8.765.2.9 sduErrorRatio . . . 649
8.765.2.10 trafficClass . . . 649
8.765.2.11 trafficPriority . . . 649
8.765.2.12 transferDelay . . . 649
8.766 wdsDhcpv4HwConfig Struct Reference . . . 649
8.766.1 Detailed Description . . . 649
8.766.2 Field Documentation . . . 649
8.766.2.1 chaddr . . . 649
8.766.2.2 chaddrLen . . . 649
8.766.2.3 hwType . . . 649
8.767 wdsDhcpv4Option Struct Reference . . . 649
8.767.1 Detailed Description . . . 650
8.767.2 Field Documentation . . . 650
8.767.2.1 optCode . . . 650
8.767.2.2 optVal . . . 650
8.767.2.3 optValLen . . . 650
8.768 wdsDhcpv4OptionList Struct Reference . . . 650
8.768.1 Detailed Description . . . 650
8.768.2 Field Documentation . . . 650
8.768.2.1 numOpt . . . 650
8.768.2.2 pOptList . . . 650
8.769 wdsDhcpv4ProfileId Struct Reference . . . 650
8.769.1 Detailed Description . . . 651
8.769.2 Field Documentation . . . 651
8.769.2.1 profileId . . . 651
8.769.2.2 profileType . . . 651

9 File Documentation **653**

9.1 apdoxypages.c File Reference . . . 653
9.1.1 Detailed Description . . . 653
9.2 common.h File Reference . . . 653
9.2.1 Macro Definition Documentation . . . 655

9.2.1.1 DEAULT_LOC_TIMEOUT_IN_SEC . . . 655
9.2.1.2 MINREQBKLEN . . . 655
9.2.1.3 MSGID_AND_LEN . . . 655
9.2.1.4 MSGID_DONT_CARE . . . 655
9.2.1.5 SDK_VALIDATE_INPUT_PACK_PARAM . . . 655
9.2.1.6 SDK_VALIDATE_INPUT_PACK_PARAM_AND_FILL_XID . . . 655
9.2.1.7 SDU_HDR_LEN . . . 655
9.2.1.8 UNUSEDPARAM . . . 655
9.2.2 Typedef Documentation . . . 655
9.2.2.1 logger . . . 655
9.2.3 Enumeration Type Documentation . . . 655
9.2.3.1 eLOG_LEVEL . . . 655
9.2.3.2 eQMI_SVC . . . 656
9.2.3.3 eTimeout . . . 656
9.2.3.4 msgtype . . . 657
9.2.4 Function Documentation . . . 657
9.2.4.1 fill_pack_ctx . . . 657
9.2.4.2 fill_sdu_hdr . . . 657
9.2.4.3 get_version . . . 657
9.2.4.4 helper_get_resp_ctx . . . 657
9.2.4.5 helper_get_xid . . . 657
9.2.4.6 helper_set_log_func . . . 657
9.2.4.7 helper_set_log_lvl . . . 657
9.2.4.8 liteqmi_GetVersion . . . 657
9.2.4.9 liteqmi_log . . . 658
9.2.4.10 unpack_result_code_only . . . 658
9.2.5 Variable Documentation . . . 658
9.2.5.1 glog . . . 658
9.2.5.2 gloglvl . . . 658
9.3 dms.h File Reference . . . 658
9.3.1 Macro Definition Documentation . . . 664
9.3.1.1 ACT_CODE_MAX_SIZE . . . 664
9.3.1.2 CK_MAX_SIZE . . . 664
9.3.1.3 DMS_IMGDETAILS_LEN . . . 664
9.3.1.4 DMS_MAX_CUST_ID_LEN . . . 664
9.3.1.5 DMS_MAX_CUST_VALUE_LEN . . . 664
9.3.1.6 DMS_MAX_FWUPDATE_LOG_STR_SZ . . . 664
9.3.1.7 DMS_MAX_FWUPDATE_REF_STR_SZ . . . 664
9.3.1.8 DMS_PM_FACTORY . . . 664
9.3.1.9 DMS_PM_LOW . . . 664

9.3.1.10 DMS_PM_OFFLINE . . . 664
9.3.1.11 DMS_PM_ONLINE . . . 664
9.3.1.12 DMS_PM_PERSISTENT_LOW . . . 664
9.3.1.13 DMS_PM_RESET . . . 664
9.3.1.14 DMS_PM_SHUT_DOWN . . . 664
9.3.1.15 DMS_SET_REPORT_DISABLE . . . 665
9.3.1.16 DMS_SET_REPORT_ENABLE . . . 665
9.3.1.17 DMS_SLQSFWINFO_APPVERSION_SZ . . . 665
9.3.1.18 DMS_SLQSFWINFO_BOOTVERSION_SZ . . . 665
9.3.1.19 DMS_SLQSFWINFO_CARRIER_SZ . . . 665
9.3.1.20 DMS_SLQSFWINFO_CUR_CARR_NAME . . . 665
9.3.1.21 DMS_SLQSFWINFO_CUR_CARR_REV . . . 665
9.3.1.22 DMS_SLQSFWINFO_MODELID_SZ . . . 665
9.3.1.23 DMS_SLQSFWINFO_PACKAGEID_SZ . . . 665
9.3.1.24 DMS_SLQSFWINFO_PRIVERSION_SZ . . . 665
9.3.1.25 DMS_SLQSFWINFO_SKU_SZ . . . 665
9.3.1.26 DMS_SWI_SET_IND_DISABLE . . . 665
9.3.1.27 DMS_SWI_SET_IND_ENABLE . . . 665
9.3.1.28 DMS_UINT8_MAX_STRING_SZ . . . 665
9.3.1.29 ERI_DATA_MAX_SIZE . . . 665
9.3.1.30 MAX_BUILD_ID_LEN . . . 665
9.3.1.31 MEID_MAX_SIZE . . . 665
9.3.1.32 SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH . . . 665
9.3.1.33 SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH . . . 665
9.3.1.34 SPC_SIZE . . . 665
9.3.1.35 UNIQUE_ID_LEN . . . 665
9.3.2 Function Documentation . . . 665
9.3.2.1 pack_dms_ActivateAutomatic . . . 665
9.3.2.2 pack_dms_GetActivationState . . . 666
9.3.2.3 pack_dms_GetBandCapability . . . 666
9.3.2.4 pack_dms_GetCrashAction . . . 666
9.3.2.5 pack_dms_GetCustFeature . . . 667
9.3.2.6 pack_dms_GetCustFeaturesV2 . . . 667
9.3.2.7 pack_dms_GetDeviceCap . . . 667
9.3.2.8 pack_dms_GetDeviceCapabilities . . . 667
9.3.2.9 pack_dms_GetDeviceHardwareRev . . . 668
9.3.2.10 pack_dms_GetDeviceMfr . . . 668
9.3.2.11 pack_dms_GetDeviceSerialNumbers . . . 669
9.3.2.12 pack_dms_GetFirmwareInfo . . . 669
9.3.2.13 pack_dms_GetFirmwareRevision . . . 669

9.3.2.14 pack_dms_GetFirmwareRevisions . . . 670
9.3.2.15 pack_dms_GetFSN . . . 670
9.3.2.16 pack_dms_GetHardwareRevision . . . 670
9.3.2.17 pack_dms_GetIMSI . . . 671
9.3.2.18 pack_dms_GetManufacturer . . . 671
9.3.2.19 pack_dms_GetModelID . . . 671
9.3.2.20 pack_dms_GetNetworkTime . . . 672
9.3.2.21 pack_dms_GetOfflineReason . . . 672
9.3.2.22 pack_dms_GetPower . . . 672
9.3.2.23 pack_dms_GetPRLVersion . . . 673
9.3.2.24 pack_dms_GetSerialNumbers . . . 673
9.3.2.25 pack_dms_GetUSBComp . . . 674
9.3.2.26 pack_dms_GetVoiceNumber . . . 674
9.3.2.27 pack_dms_ResetToFactoryDefaults . . . 674
9.3.2.28 pack_dms_SetActivationStatusCallback . . . 675
9.3.2.29 pack_dms_SetCrashAction . . . 675
9.3.2.30 pack_dms_SetCustFeature . . . 675
9.3.2.31 pack_dms_SetCustFeaturesV2 . . . 676
9.3.2.32 pack_dms_SetEventReport . . . 676
9.3.2.33 pack_dms_SetFirmwarePreference . . . 676
9.3.2.34 pack_dms_SetPower . . . 677
9.3.2.35 pack_dms_SetUSBComp . . . 677
9.3.2.36 pack_dms_SLQSDmsSwiGetResetInfo . . . 678
9.3.2.37 pack_dms_SLQSDmsSwiIndicationRegister . . . 678
9.3.2.38 pack_dms_SLQSGetBandCapability . . . 678
9.3.2.39 pack_dms_SLQSGetERIFile . . . 679
9.3.2.40 pack_dms_SLQSSwiClearDyingGaspStatistics . . . 679
9.3.2.41 pack_dms_SLQSSwiGetCrashInfo . . . 679
9.3.2.42 pack_dms_SLQSSwiGetDyingGaspCfg . . . 680
9.3.2.43 pack_dms_SLQSSwiGetDyingGaspStatistics . . . 680
9.3.2.44 pack_dms_SLQSSwiGetFirmwareCurr . . . 680
9.3.2.45 pack_dms_SLQSSwiGetFwUpdateStatus . . . 681
9.3.2.46 pack_dms_SLQSSwiGetHostDevInfo . . . 681
9.3.2.47 pack_dms_SLQSSwiGetOSInfo . . . 681
9.3.2.48 pack_dms_SLQSSwiGetSerialNoExt . . . 682
9.3.2.49 pack_dms_SLQSSwiSetDyingGaspCfg . . . 682
9.3.2.50 pack_dms_SLQSSwiSetHostDevInfo . . . 682
9.3.2.51 pack_dms_SLQSSwiSetOSInfo . . . 683
9.3.2.52 pack_dms_SLQSUIMGetState . . . 683
9.3.2.53 pack_dms_UIMChangePIN . . . 683

9.3.2.54 pack_dms_UIMGetControlKeyStatus . . . 684
9.3.2.55 pack_dms_UIMGetICCID . . . 684
9.3.2.56 pack_dms_UIMGetPINStatus . . . 685
9.3.2.57 pack_dms_UIMSetControlKeyProtection . . . 685
9.3.2.58 pack_dms_UIMSetPINProtection . . . 685
9.3.2.59 pack_dms_UIMUnblockControlKey . . . 686
9.3.2.60 pack_dms_UIMUnblockPIN . . . 686
9.3.2.61 pack_dms_UIMVerifyPIN . . . 686
9.3.2.62 pack_dms_ValidateSPC . . . 687
9.3.2.63 unpack_dms_ActivateAutomatic . . . 687
9.3.2.64 unpack_dms_GetActivationState . . . 688
9.3.2.65 unpack_dms_GetBandCapability . . . 688
9.3.2.66 unpack_dms_GetCrashAction . . . 688
9.3.2.67 unpack_dms_GetCustFeature . . . 689
9.3.2.68 unpack_dms_GetCustFeaturesV2 . . . 689
9.3.2.69 unpack_dms_GetDeviceCap . . . 689
9.3.2.70 unpack_dms_GetDeviceCapabilities . . . 689
9.3.2.71 unpack_dms_GetDeviceHardwareRev . . . 690
9.3.2.72 unpack_dms_GetDeviceMfr . . . 690
9.3.2.73 unpack_dms_GetDeviceSerialNumbers . . . 690
9.3.2.74 unpack_dms_GetFirmwareInfo . . . 691
9.3.2.75 unpack_dms_GetFirmwareRevision . . . 691
9.3.2.76 unpack_dms_GetFirmwareRevisions . . . 691
9.3.2.77 unpack_dms_GetFSN . . . 692
9.3.2.78 unpack_dms_GetHardwareRevision . . . 692
9.3.2.79 unpack_dms_GetIMSI . . . 692
9.3.2.80 unpack_dms_GetManufacturer . . . 693
9.3.2.81 unpack_dms_GetModelID . . . 693
9.3.2.82 unpack_dms_GetNetworkTime . . . 693
9.3.2.83 unpack_dms_GetOfflineReason . . . 694
9.3.2.84 unpack_dms_GetPower . . . 694
9.3.2.85 unpack_dms_GetPRLVersion . . . 694
9.3.2.86 unpack_dms_GetSerialNumbers . . . 695
9.3.2.87 unpack_dms_GetUSBComp . . . 695
9.3.2.88 unpack_dms_GetVoiceNumber . . . 695
9.3.2.89 unpack_dms_ResetToFactoryDefaults . . . 696
9.3.2.90 unpack_dms_SetActivationStatusCallback . . . 696
9.3.2.91 unpack_dms_SetCrashAction . . . 696
9.3.2.92 unpack_dms_SetCustFeature . . . 697
9.3.2.93 unpack_dms_SetCustFeaturesV2 . . . 697

9.3.2.94 unpack_dms_SetEventReport . . . 697
9.3.2.95 unpack_dms_SetEventReport_ind . . . 698
9.3.2.96 unpack_dms_SetFirmwarePreference . . . 698
9.3.2.97 unpack_dms_SetPower . . . 698
9.3.2.98 unpack_dms_SetUSBComp . . . 699
9.3.2.99 unpack_dms_SLQSDmsSwiGetResetInfo . . . 699
9.3.2.100 unpack_dms_SLQSDmsSwiGetResetInfo_Ind . . . 699
9.3.2.101 unpack_dms_SLQSDmsSwiIndicationRegister . . . 700
9.3.2.102 unpack_dms_SLQSGetBandCapability . . . 700
9.3.2.103 unpack_dms_SLQSGetERIFile . . . 701
9.3.2.104 unpack_dms_SLQSSwiClearDyingGaspStatistics . . . 701
9.3.2.105 unpack_dms_SLQSSwiGetCrashInfo . . . 701
9.3.2.106 unpack_dms_SLQSSwiGetDyingGaspCfg . . . 702
9.3.2.107 unpack_dms_SLQSSwiGetDyingGaspStatistics . . . 702
9.3.2.108 unpack_dms_SLQSSwiGetFirmwareCurr . . . 702
9.3.2.109 unpack_dms_SLQSSwiGetFwUpdateStatus . . . 703
9.3.2.110 unpack_dms_SLQSSwiGetHostDevInfo . . . 703
9.3.2.111 unpack_dms_SLQSSwiGetOSInfo . . . 703
9.3.2.112 unpack_dms_SLQSSwiGetSerialNoExt . . . 704
9.3.2.113 unpack_dms_SLQSSwiSetDyingGaspCfg . . . 704
9.3.2.114 unpack_dms_SLQSSwiSetHostDevInfo . . . 704
9.3.2.115 unpack_dms_SLQSSwiSetOSInfo . . . 705
9.3.2.116 unpack_dms_SLQSUIMGetState . . . 705
9.3.2.117 unpack_dms_UIMChangePIN . . . 705
9.3.2.118 unpack_dms_UIMGetControlKeyStatus . . . 706
9.3.2.119 unpack_dms_UIMGetICCID . . . 706
9.3.2.120 unpack_dms_UIMGetPINStatus . . . 706
9.3.2.121 unpack_dms_UIMSetControlKeyProtection . . . 707
9.3.2.122 unpack_dms_UIMSetPINProtection . . . 707
9.3.2.123 unpack_dms_UIMUnblockControlKey . . . 707
9.3.2.124 unpack_dms_UIMUnblockPIN . . . 708
9.3.2.125 unpack_dms_UIMVerifyPIN . . . 708
9.3.2.126 unpack_dms_ValidateSPC . . . 708
9.4 fms.h File Reference . . . 709
9.4.1 Macro Definition Documentation . . . 710
9.4.1.1 FMS_FW_PRI_BUILD_MATCH_LEN . . . 710
9.4.1.2 FMS_GOBI_LISTENTRIES_MAX . . . 710
9.4.1.3 FMS_GOBI_MBN_BUILD_ID_STR_LEN . . . 710
9.4.1.4 FMS_GOBI_MBN_IMG_ID_STR_LEN . . . 710
9.4.1.5 FMS_IMAGE_ID_BUILD_ID_LEN . . . 710

9.4.1.6 FMS_IMAGE_ID_IMG_ID_LEN . . . 710
9.4.1.7 FMS_IMAGE_ID_MAX_ENTRIES . . . 710
9.4.1.8 FMS_IMAGE_ID_PRI_IMGTYPE . . . 710
9.4.1.9 FMS_MAX_IMAGE_ID_ELEMENT . . . 710
9.4.1.10 FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE . . . 710
9.4.2 Function Documentation . . . 710
9.4.2.1 GetValidFwPriCombinations . . . 710
9.4.2.2 pack_fms_GetImagesPreference . . . 710
9.4.2.3 pack_fms_GetStoredImages . . . 711
9.4.2.4 pack_fms_SetImagesPreference . . . 711
9.4.2.5 unpack_fms_GetImagesPreference . . . 711
9.4.2.6 unpack_fms_GetStoredImages . . . 711
9.4.2.7 unpack_fms_SetImagesPreference . . . 712
9.5 lite-fw.h File Reference . . . 712
9.5.1 Detailed Description . . . 713
9.5.2 Macro Definition Documentation . . . 713
9.5.2.1 FIRMWARE_INFO_STRING_SIZE . . . 713
9.5.2.2 IMG_MASK_CLEAR . . . 714
9.5.2.3 IMG_MASK_GENERIC . . . 714
9.5.2.4 IMG_MASK_MDM . . . 714
9.5.2.5 IMG_MASK_PRI . . . 714
9.5.2.6 libSDP_BuildImagesPreferenceRequest . . . 714
9.5.2.7 libSDP_CalculateImageMask . . . 714
9.5.2.8 LIBSDP_CARRIER_PACKAGE_SKU . . . 714
9.5.2.9 libSDP_CheckValidFirmwareInfo . . . 714
9.5.2.10 libSDP_DownloadFW . . . 714
9.5.2.11 libSDP_ExtractFirmwareParametersByPath . . . 714
9.5.2.12 libSDP_getFileType . . . 714
9.5.2.13 libSDP_GetModelFamily . . . 714
9.5.2.14 libSDP_GetVersion . . . 714
9.5.2.15 libsdp_set_log_func . . . 714
9.5.2.16 libsdp_SetReadBlockSize . . . 714
9.5.2.17 LIBSDP_SKU_STRING_LENGTH . . . 714
9.5.2.18 LITEFW_CARRIER_PACKAGE_SKU . . . 714
9.5.2.19 LITEFW_SKU_STRING_LENGTH . . . 714
9.5.3 Typedef Documentation . . . 714
9.5.3.1 libSDP_FirmwareInfo . . . 714
9.5.3.2 libsdplogger . . . 714
9.5.3.3 litefw_FirmwareInfo . . . 714
9.5.3.4 litefwlogger . . . 715

9.5.4 Enumeration Type Documentation . . . 715
9.5.4.1 litefw_Fw_Type . . . 715
9.5.4.2 litefw_fwdwl_error_codes . . . 715
9.5.4.3 litefw_Models . . . 716
9.5.5 Function Documentation . . . 716
9.5.5.1 litefw_BuildImagesPreferenceRequest . . . 716
9.5.5.2 litefw_CalculateImageMask . . . 716
9.5.5.3 litefw_CheckValidFirmwareInfo . . . 717
9.5.5.4 litefw_DownloadFW . . . 717
9.5.5.5 litefw_ExtractFirmwareParametersByPath . . . 718
9.5.5.6 litefw_getFileType . . . 718
9.5.5.7 litefw_GetModelFamily . . . 718
9.5.5.8 litefw_GetVersion . . . 719
9.5.5.9 litefw_set_log_func . . . 719
9.5.5.10 litefw_SetReadBlockSize . . . 719
9.6 loc.h File Reference . . . 719
9.6.1 Macro Definition Documentation . . . 722
9.6.1.1 LOC_UINT8_MAX_STRING_SZ . . . 722
9.6.1.2 LOCEVENTMASKBATCHFULLNOTIFICATION . . . 722
9.6.1.3 LOCEVENTMASKENGINESTATE . . . 722
9.6.1.4 LOCEVENTMASKFIXSESSIONSTATE . . . 722
9.6.1.5 LOCEVENTMASKGEOFENCEBATCHBREACHNOTIFICATION . . . 722
9.6.1.6 LOCEVENTMASKGEOFENCEBREACHNOTIFICATION . . . 722
9.6.1.7 LOCEVENTMASKGEOFENCEGENALERT . . . 722
9.6.1.8 LOCEVENTMASKGNSSMEASUREMENTREPORT . . . 723
9.6.1.9 LOCEVENTMASKGNSSSVINFO . . . 723
9.6.1.10 LOCEVENTMASKINJECTPOSITIONREQ . . . 723
9.6.1.11 LOCEVENTMASKINJECTPREDICTEDORBITSREQ . . . 723
9.6.1.12 LOCEVENTMASKINJECTTIMEREQ . . . 723
9.6.1.13 LOCEVENTMASKINJECTWIFIAPDATAREQ . . . 723
9.6.1.14 LOCEVENTMASKINVALIDVALUE . . . 723
9.6.1.15 LOCEVENTMASKLIVEBATCHEDPOSITIONREPORT . . . 723
9.6.1.16 LOCEVENTMASKLOCATIONSERVERCONNECTIONREQ . . . 723
9.6.1.17 LOCEVENTMASKMOTIONDATACONTROL . . . 723
9.6.1.18 LOCEVENTMASKNIGEOFENCENOTIFICATION . . . 723
9.6.1.19 LOCEVENTMASKNINOTIFYVERIFYREQ . . . 724
9.6.1.20 LOCEVENTMASKNMEA . . . 724
9.6.1.21 LOCEVENTMASKPEDOMETERCONTROL . . . 724
9.6.1.22 LOCEVENTMASKPOSITIONREPORT . . . 724
9.6.1.23 LOCEVENTMASKSENSORSTREAMINGREADYSTATUS . . . 724

9.6.1.24 LOCEVENTMASKSETSPISTREAMINGREPORT . . . 724
9.6.1.25 LOCEVENTMASKTIMESYNCREQ . . . 724
9.6.1.26 LOCEVENTMASKVEHICLEDATAREADYSTATUS . . . 724
9.6.1.27 LOCEVENTMASKWIFIREQ . . . 724
9.6.1.28 MAX_SENSOR_DATA_LEN . . . 724
9.6.1.29 MAX_TEMP_DATA_LEN . . . 724
9.6.2 Enumeration Type Documentation . . . 724
9.6.2.1 anonymous enum . . . 724
9.6.3 Function Documentation . . . 725
9.6.3.1 pack_loc_DeleteAssistData . . . 725
9.6.3.2 pack_loc_EventRegister . . . 725
9.6.3.3 pack_loc_SetExtPowerState . . . 725
9.6.3.4 pack_loc_SetOperationMode . . . 726
9.6.3.5 pack_loc_SLQSLOCGetBestAvailPos . . . 726
9.6.3.6 pack_loc_SLQSLOCInjectPosition . . . 726
9.6.3.7 pack_loc_SLQSLOCInjectSensorData . . . 727
9.6.3.8 pack_loc_SLQSLOCInjectUTCTime . . . 727
9.6.3.9 pack_loc_SLQSLOCSetCradleMountConfig . . . 727
9.6.3.10 pack_loc_Start . . . 728
9.6.3.11 pack_loc_Stop . . . 728
9.6.3.12 unpack_loc_BestAvailPos_Ind . . . 728
9.6.3.13 unpack_loc_DeleteAssistData . . . 729
9.6.3.14 unpack_loc_DeleteAssistData_Ind . . . 729
9.6.3.15 unpack_loc_EngineState_Ind . . . 729
9.6.3.16 unpack_loc_EventRegister . . . 730
9.6.3.17 unpack_loc_GnssSvInfo_Ind . . . 730
9.6.3.18 unpack_loc_PositionRpt_Ind . . . 730
9.6.3.19 unpack_loc_SetExtPowerConfig_Ind . . . 731
9.6.3.20 unpack_loc_SetExtPowerState . . . 731
9.6.3.21 unpack_loc_SetOperationMode . . . 731
9.6.3.22 unpack_loc_SetOperationMode_Ind . . . 732
9.6.3.23 unpack_loc_SLQSLOCGetBestAvailPos . . . 732
9.6.3.24 unpack_loc_SLQSLOCInjectPosition . . . 732
9.6.3.25 unpack_loc_SLQSLOCInjectSensorData . . . 733
9.6.3.26 unpack_loc_SLQSLOCInjectUTCTime . . . 733
9.6.3.27 unpack_loc_SLQSLOCSetCradleMountConfig . . . 733
9.6.3.28 unpack_loc_Start . . . 734
9.6.3.29 unpack_loc_Stop . . . 734
9.7 nas.h File Reference . . . 734
9.7.1 Macro Definition Documentation . . . 742

9.7.1.1 NAS_IMSI_M_S1_LENGTH . . . 742
9.7.1.2 NAS_IMSI_M_S2_LENGTH . . . 742
9.7.1.3 NAS_MAX_DESCRIPTION_LENGTH . . . 742
9.7.1.4 NAS_MAX_NUM_NETWORKS . . . 742
9.7.1.5 NAS_MAX_SCC_RX_INFO_INSTANCES . . . 742
9.7.1.6 NAS_NAM_NAME_LENGTH . . . 742
9.7.1.7 NAS_OTA_MESSAGE_MAX_BUF_SIZE . . . 742
9.7.1.8 NAS_PLMN_LENGTH . . . 742
9.7.1.9 NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST . . . 742
9.7.2 Enumeration Type Documentation . . . 742
9.7.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB . . . 742
9.7.2.2 LITEQMI_NAS_LTE_CPHY_SCELL_STATE . . . 743
9.7.2.3 NAS_LTE_CPHY_CA_BW_NRB_LITE . . . 743
9.7.2.4 NAS_LTE_CPHY_SCELL_STATE_LITE . . . 743
9.7.3 Function Documentation . . . 743
9.7.3.1 pack_nas_GetACCOLC . . . 743
9.7.3.2 pack_nas_GetANAAAAuthenticationStatus . . . 743
9.7.3.3 pack_nas_GetCDMANetworkParameters . . . 744
9.7.3.4 pack_nas_GetHomeNetwork . . . 744
9.7.3.5 pack_nas_GetHomeNetwork3GPP2 . . . 744
9.7.3.6 pack_nas_GetNetworkPreference . . . 745
9.7.3.7 pack_nas_GetRFInfo . . . 745
9.7.3.8 pack_nas_GetServingNetwork . . . 745
9.7.3.9 pack_nas_GetServingNetworkCapabilities . . . 745
9.7.3.10 pack_nas_GetSignalStrengths . . . 746
9.7.3.11 pack_nas_InitiateDomainAttach . . . 746
9.7.3.12 pack_nas_PerformNetworkScan . . . 746
9.7.3.13 pack_nas_SetACCOLC . . . 747
9.7.3.14 pack_nas_SetCDMANetworkParameters . . . 747
9.7.3.15 pack_nas_SetLURejectCallback . . . 747
9.7.3.16 pack_nas_SetNetworkPreference . . . 748
9.7.3.17 pack_nas_SetRFInfoCallback . . . 748
9.7.3.18 pack_nas_SLQSConfigSigInfo . . . 748
9.7.3.19 pack_nas_SLQSGetErrorRate . . . 749
9.7.3.20 pack_nas_SlqsGetLTECphyCAInfo . . . 749
9.7.3.21 pack_nas_SLQSGetNetworkTime . . . 749
9.7.3.22 pack_nas_SLQSGetOperatorNameData . . . 749
9.7.3.23 pack_nas_SLQSGetPLMNName . . . 750
9.7.3.24 pack_nas_SLQSGetServingSystem . . . 750
9.7.3.25 pack_nas_SLQSGetSignalStrength . . . 750

9.7.3.26 pack_nas_SLQSGetSysInfo . . . 751
9.7.3.27 pack_nas_SLQSGetSysSelectionPref . . . 751
9.7.3.28 pack_nas_SLQSInitiateNetworkRegistration . . . 751
9.7.3.29 pack_nas_SLQSNasConfigSigInfo2 . . . 752
9.7.3.30 pack_nas_SLQSNasGet3GPP2Subscription . . . 752
9.7.3.31 pack_nas_SLQSNasGetCellLocationInfo . . . 752
9.7.3.32 pack_nas_SLQSNasGetHDRColorCode . . . 753
9.7.3.33 pack_nas_SLQSNasGetSigInfo . . . 753
9.7.3.34 pack_nas_SLQSNasGetTxRxInfo . . . 753
9.7.3.35 pack_nas_SLQSNasIndicationRegisterExt . . . 754
9.7.3.36 pack_nas_SLQSNASSwiGetChannelLock . . . 754
9.7.3.37 pack_nas_SLQSNasSwiIndicationRegister . . . 754
9.7.3.38 pack_nas_SLQSNasSwiModemStatus . . . 755
9.7.3.39 pack_nas_SLQSNASSwiSetChannelLock . . . 755
9.7.3.40 pack_nas_SLQSSetBandPreference . . . 755
9.7.3.41 pack_nas_SLQSSetSignalStrengthsCallback . . . 756
9.7.3.42 pack_nas_SLQSSetSysSelectionPref . . . 756
9.7.3.43 pack_nas_SLQSSwiGetHDRPersonality . . . 756
9.7.3.44 pack_nas_SLQSSwiGetHDRProtSubtype . . . 757
9.7.3.45 pack_nas_SLQSSwiGetHRPDStats . . . 757
9.7.3.46 pack_nas_SLQSSwiGetLteCQI . . . 757
9.7.3.47 pack_nas_SLQSSwiGetLteSccRxInfo . . . 758
9.7.3.48 pack_nas_SLQSSwiNetworkDebug . . . 758
9.7.3.49 pack_nas_SLQSSwiPSDetach . . . 758
9.7.3.50 unpack_nas_GetACCOLC . . . 759
9.7.3.51 unpack_nas_GetANAAAAuthenticationStatus . . . 759
9.7.3.52 unpack_nas_GetCDMANetworkParameters . . . 759
9.7.3.53 unpack_nas_GetHomeNetwork . . . 760
9.7.3.54 unpack_nas_GetHomeNetwork3GPP2 . . . 760
9.7.3.55 unpack_nas_GetNetworkPreference . . . 760
9.7.3.56 unpack_nas_GetRFInfo . . . 760
9.7.3.57 unpack_nas_GetServingNetwork . . . 761
9.7.3.58 unpack_nas_GetServingNetworkCapabilities . . . 761
9.7.3.59 unpack_nas_GetSignalStrengths . . . 761
9.7.3.60 unpack_nas_InitiateDomainAttach . . . 762
9.7.3.61 unpack_nas_PerformNetworkScan . . . 762
9.7.3.62 unpack_nas_SetACCOLC . . . 762
9.7.3.63 unpack_nas_SetCDMANetworkParameters . . . 763
9.7.3.64 unpack_nas_SetDataCapabilitiesCallback_ind . . . 763
9.7.3.65 unpack_nas_SetEventReportInd . . . 763

9.7.3.66 unpack_nas_SetLURejectCallback . . . 763
9.7.3.67 unpack_nas_SetNasLTECphyCaIndCallback_ind . . . 764
9.7.3.68 unpack_nas_SetNetworkPreference . . . 764
9.7.3.69 unpack_nas_SetRFInfoCallback . . . 764
9.7.3.70 unpack_nas_SetRoamingIndicatorCallback_ind . . . 764
9.7.3.71 unpack_nas_SetServingSystemCallback_ind . . . 765
9.7.3.72 unpack_nas_SLQSConfigSigInfo . . . 765
9.7.3.73 unpack_nas_SLQSGetErrorRate . . . 765
9.7.3.74 unpack_nas_SlqsGetLTECphyCAInfo . . . 765
9.7.3.75 unpack_nas_SLQSGetNetworkTime . . . 765
9.7.3.76 unpack_nas_SLQSGetOperatorNameData . . . 766
9.7.3.77 unpack_nas_SLQSGetPLMNName . . . 766
9.7.3.78 unpack_nas_SLQSGetServingSystem . . . 766
9.7.3.79 unpack_nas_SLQSGetSignalStrength . . . 767
9.7.3.80 unpack_nas_SLQSGetSysInfo . . . 767
9.7.3.81 unpack_nas_SLQSGetSysSelectionPref . . . 767
9.7.3.82 unpack_nas_SLQSInitiateNetworkRegistration . . . 768
9.7.3.83 unpack_nas_SLQSNasConfigSigInfo2 . . . 768
9.7.3.84 unpack_nas_SLQSNasGet3GPP2Subscription . . . 768
9.7.3.85 unpack_nas_SLQSNasGetCellLocationInfo . . . 769
9.7.3.86 unpack_nas_SLQSNasGetHDRColorCode . . . 769
9.7.3.87 unpack_nas_SLQSNasGetSigInfo . . . 769
9.7.3.88 unpack_nas_SLQSNasGetTxRxInfo . . . 770
9.7.3.89 unpack_nas_SLQSNasIndicationRegisterExt . . . 770
9.7.3.90 unpack_nas_SLQSNasNetworkTimeCallBack_ind . . . 770
9.7.3.91 unpack_nas_SLQSNasSigInfoCallback_ind . . . 770
9.7.3.92 unpack_nas_SLQSNASSwiGetChannelLock . . . 771
9.7.3.93 unpack_nas_SLQSNasSwiIndicationRegister . . . 771
9.7.3.94 unpack_nas_SLQSNasSwiModemStatus . . . 771
9.7.3.95 unpack_nas_SLQSNasSwiOTAMessageCallback_ind . . . 772
9.7.3.96 unpack_nas_SLQSNASSwiSetChannelLock . . . 772
9.7.3.97 unpack_nas_SLQSNasSysInfoCallback_ind . . . 772
9.7.3.98 unpack_nas_SLQSNasTimerCallback_ind . . . 773
9.7.3.99 unpack_nas_SLQSSetBandPreference . . . 773
9.7.3.100 unpack_nas_SLQSSetSignalStrengthsCallback . . . 773
9.7.3.101 unpack_nas_SLQSSetSysSelectionPref . . . 773
9.7.3.102 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind . . . 774
9.7.3.103 unpack_nas_SLQSSwiGetHDRPersonality . . . 774
9.7.3.104 unpack_nas_SLQSSwiGetHDRProtSubtype . . . 774
9.7.3.105 unpack_nas_SLQSSwiGetHRPDStats . . . 775

9.7.3.106 unpack_nas_SLQSSwiGetLteCQI . . . 775
9.7.3.107 unpack_nas_SLQSSwiGetLteSccRxInfo . . . 775
9.7.3.108 unpack_nas_SLQSSwiNetworkDebug . . . 776
9.7.3.109 unpack_nas_SLQSSwiPSDetach . . . 776
9.8 pds.h File Reference . . . 776
9.8.1 Function Documentation . . . 777
9.8.1.1 pack_pds_GetPDSDefaults . . . 777
9.8.1.2 pack_pds_GetPDSState . . . 777
9.8.1.3 pack_pds_GetPortAutomaticTracking . . . 778
9.8.1.4 pack_pds_SetPDSDefaults . . . 778
9.8.1.5 pack_pds_SetPDSState . . . 778
9.8.1.6 pack_pds_SetPortAutomaticTracking . . . 779
9.8.1.7 unpack_pds_GetPDSDefaults . . . 779
9.8.1.8 unpack_pds_GetPDSState . . . 779
9.8.1.9 unpack_pds_GetPortAutomaticTracking . . . 780
9.8.1.10 unpack_pds_SetPDSDefaults . . . 780
9.8.1.11 unpack_pds_SetPDSState . . . 780
9.8.1.12 unpack_pds_SetPortAutomaticTracking . . . 781
9.9 qaGobiApiTableBandClasses.h File Reference . . . 781
9.9.1 Detailed Description . . . 781
9.9.2 Band Classes (Value - Description) . . . 781
9.9.2.1 LTE Bands . . . 783
9.10 qaGobiApiTableCallControlReturnReasons.h File Reference . . . 784
9.10.1 Detailed Description . . . 784
9.10.2 Call Control Result Reasons (Value - Name - Description) . . . 784
9.11 qaGobiApiTableCallEndReasons.h File Reference . . . 785
9.11.1 Detailed Description . . . 785
9.11.2 Call end reason codes (Code - Reason) . . . 785
9.11.2.1 Technology-agnostic call end reasons . . . 785
9.11.2.2 EVDO CDMA 1xEV-DO . . . 785
9.11.2.3 WCDMA/GSM call end reasons . . . 786
9.11.2.4 EVDO CDMA 1xEV-DO . . . 788
9.11.2.5 call end reason type . . . 789
9.11.2.6 Mobile IP call end reasons (Type=1) . . . 789
9.11.2.7 Internal call end reasons (Type=2) . . . 791
9.11.2.8 Call Manager defined call end reasons (Type=3) . . . 792
9.11.2.9 3GPP specification defined call end reasons (Type=6) . . . 797
9.11.2.10 PPP call end reasons (Type=7) . . . 799
9.11.2.11 EHRPD call end reasons (Type=8) . . . 799
9.11.2.12 IPV6 call end reasons (Type=9) . . . 800

9.12 qaGobiApiTableCarrierCodes.h File Reference . . . 800
9.12.1 Detailed Description . . . 800
9.12.2 Carrier Codes (Number - Carrier) . . . 800
9.13 qaGobiApiTableCodingScheme.h File Reference . . . 802
9.13.1 Detailed Description . . . 802
9.13.2 Call Control Result Reasons (Value - Name - Description) . . . 802
9.13.2.1 Use of bits 3..0\n\n . . . 802
9.13.3 Coding Group Bits 7..4(0001) . . . 803
9.13.3.1 use of bits 3..0 . . . 803
9.13.4 Coding Group Bits 7..4(0010) . . . 803
9.13.4.1 use of bits 3..0 . . . 803
9.13.5 Coding Group Bits 7..4(0011) . . . 803
9.13.5.1 use of bits 3..0 . . . 803
9.13.6 Coding Group Bits 7..4(01xx) . . . 804
9.13.6.1 use of bits 3..0 . . . 804
9.13.7 Coding Group Bits 7..4(1001) . . . 804
9.13.7.1 Reserved coding groups . . . 804
9.13.8 Coding Group Bits 7..4(1010..1101) . . . 804
9.13.8.1 Reserved coding groups . . . 804
9.13.9 Coding Group Bits 7..4(1110) . . . 804
9.13.9.1 Defined by the WAP Forum . . . 804
9.13.10 Coding Group Bits 7..4 (1111) . . . 804
9.13.10.1 Data coding / message handling . . . 805
9.13.11 Macro Definition Documentation . . . 805
9.13.11.1 __GOBI_API_CODING_SCHEME_H__ . . . 805
9.14 qaGobiApiTableGpsCapabilityCodes.h File Reference . . . 805
9.14.1 Detailed Description . . . 805
9.14.2 GPS capability (Value - Capability) . . . 805
9.15 qaGobiApiTablePowerModes.h File Reference . . . 805
9.15.1 Detailed Description . . . 805
9.15.2 Power Modes (Value - Description) . . . 806
9.16 qaGobiApiTableRadioInterfaces.h File Reference . . . 806
9.16.1 Detailed Description . . . 806
9.16.2 Radio interface . . . 806
9.16.2.1 Technology (Value - Radio Interface Technology) . . . 806
9.17 qaGobiApiTableRegionCodes.h File Reference . . . 807
9.17.1 Detailed Description . . . 807
9.17.2 Region Codes (Code - Region) . . . 807
9.18 qaGobiApiTableServiceOptions.h File Reference . . . 807
9.18.1 Detailed Description . . . 807

9.18.2 Service Option codes (Code - Reason) . . . 807
9.18.2.1 Description . . . 807
9.19 qaGobiApiTableSupServiceInfoClasses.h File Reference . . . 809
9.19.1 Detailed Description . . . 809
9.19.2 Supplementary Service Information Classes (Value - Service Class) . . . 810
9.20 qaGobiApiTableSwiAudio.h File Reference . . . 810
9.20.1 Detailed Description . . . 810
9.20.2 ACDB Device (Device ID - description) . . . 810
9.20.3 Physical Interface (Device ID - description - Interface parameters) . . . 810
9.21 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference . . . 810
9.21.1 Detailed Description . . . 811
9.21.2 OMA DM Update Complete Status (Update Complete Status - Meaning - Usage) . . . 811
9.22 qaGobiApiTableVoiceCallEndReasons.h File Reference . . . 812
9.22.1 Detailed Description . . . 812
9.22.2 Voice Call and supplementary services end reason codes (Code - Reason) . . . 812
9.22.2.1 General . . . 812
9.22.2.2 service Errors . . . 814
9.22.2.3 control cause values . . . 815
9.22.2.4 reject causes . . . 816
9.22.2.5 reject causes . . . 817
9.22.2.6 reject causes . . . 817
9.22.2.7 stratum reject causes . . . 817
9.22.2.8 reject causes . . . 818
9.22.2.9 IP end reasons . . . 818
9.23 qmerrno.h File Reference . . . 818
9.23.1 Enumeration Type Documentation . . . 820
9.23.1.1 eQCWWANError . . . 820
9.23.1.2 qm_wds_ds_profile_extended_err_codes . . . 825
9.24 qos.h File Reference . . . 825
9.24.1 Macro Definition Documentation . . . 827
9.24.1.1 LITEQMI_MAX_QOS_FILTERS . . . 827
9.24.1.2 LITEQMI_MAX_QOS_FLOW_PER_APN_STATS . . . 827
9.24.1.3 LITEQMI_MAX_QOS_FLOWS . . . 827
9.24.2 Function Documentation . . . 827
9.24.2.1 pack_qos_SLQSQosGetNetworkStatus . . . 827
9.24.2.2 pack_qos_SLQSQosSwiReadApnExtraParams . . . 827
9.24.2.3 pack_qos_SLQSQosSwiReadDataStats . . . 828
9.24.2.4 pack_qos_SLQSSetQosEventCallback . . . 829
9.24.2.5 unpack_qos_SLQSQosGetNetworkStatus . . . 829
9.24.2.6 unpack_qos_SLQSQosSwiReadApnExtraParams . . . 830

9.24.2.7 unpack_qos_SLQSQosSwiReadDataStats . . . 830
9.24.2.8 unpack_qos_SLQSSetQosEventCallback . . . 831
9.24.2.9 unpack_qos_SLQSSetQosEventCallback_ind . . . 831
9.24.2.10 unpack_qos_SLQSSetQosNWStatusCallback_ind . . . 832
9.24.2.11 unpack_qos_SLQSSetQosPriEventCallback_ind . . . 832
9.24.2.12 unpack_qos_SLQSSetQosStatusCallback_ind . . . 833
9.25 rms.h File Reference . . . 833
9.25.1 Function Documentation . . . 834
9.25.1.1 pack_rms_GetSMSWake . . . 834
9.25.1.2 pack_rms_SetSMSWake . . . 834
9.25.1.3 unpack_rms_GetSMSWake . . . 834
9.25.1.4 unpack_rms_SetSMSWake . . . 835
9.26 sar.h File Reference . . . 835
9.26.1 Function Documentation . . . 835
9.26.1.1 pack_sar_SLQSGetRfSarState . . . 835
9.26.1.2 pack_sar_SLQSSetRfSarState . . . 836
9.26.1.3 unpack_sar_SLQSGetRfSarState . . . 836
9.26.1.4 unpack_sar_SLQSSetRfSarState . . . 837
9.27 sms.h File Reference . . . 837
9.27.1 Macro Definition Documentation . . . 840
9.27.1.1 MAX_CDMA_ENC_MO_TXT_MSG_SIZE . . . 840
9.27.1.2 MAX_MS_TRANSFER_ROUTE_MSG . . . 840
9.27.1.3 MAX_MSC_ADDRESS_SIZE . . . 840
9.27.1.4 MAX_MSE_TWS_MSG . . . 840
9.27.1.5 MAX_SMS_LIST_SIZE . . . 840
9.27.1.6 MAX_SMS_MESSAGE_SIZE . . . 840
9.27.1.7 SMS_CONFIG_LEN . . . 840
9.27.1.8 SMS_MAX_SMS_ROUTES . . . 840
9.27.1.9 SMS_NUM_OF_SET . . . 840
9.27.1.10 SMSC_TYPE_LEN . . . 840
9.27.2 Typedef Documentation . . . 841
9.27.2.1 sMSCAddressInfo . . . 841
9.27.2.2 sMSEtwsMessageInfo . . . 841
9.27.2.3 sMSEtwsPlmnInfo . . . 841
9.27.2.4 sMSMessageModeInfo . . . 841
9.27.2.5 sMSMTMessageInfo . . . 841
9.27.2.6 sMSOnIMSInfo . . . 841
9.27.2.7 sMSTransferRouteMTMessageInfo . . . 842
9.27.3 Enumeration Type Documentation . . . 842
9.27.3.1 eqmiCbkSetStatus . . . 842

9.27.4 Function Documentation . . . 842
9.27.4.1 pack_sms_GetSMSCAddress . . . 842
9.27.4.2 pack_sms_SaveSMS . . . 843
9.27.4.3 pack_sms_SendSMS . . . 843
9.27.4.4 pack_sms_SetNewSMSCallback . . . 843
9.27.4.5 pack_sms_SetSMSCAddress . . . 844
9.27.4.6 pack_sms_SLQSDeleteSMS . . . 844
9.27.4.7 pack_sms_SLQSGetIndicationRegister . . . 844
9.27.4.8 pack_sms_SLQSGetMessageWaiting . . . 845
9.27.4.9 pack_sms_SLQSGetSMS . . . 845
9.27.4.10 pack_sms_SLQSGetSmsBroadcastConfig . . . 846
9.27.4.11 pack_sms_SLQSGetSMSList . . . 846
9.27.4.12 pack_sms_SLQSGetTransLayerInfo . . . 846
9.27.4.13 pack_sms_SLQSGetTransNWRegInfo . . . 847
9.27.4.14 pack_sms_SLQSModifySMSStatus . . . 847
9.27.4.15 pack_sms_SLQSSendAsyncSMS . . . 847
9.27.4.16 pack_sms_SLQSSetIndicationRegister . . . 848
9.27.4.17 pack_sms_SLQSSetSmsBroadcastActivation . . . 848
9.27.4.18 pack_sms_SLQSSetSmsBroadcastConfig . . . 848
9.27.4.19 pack_sms_SLQSSetSmsStorage . . . 849
9.27.4.20 pack_sms_SLQSSmsGetMaxStorageSize . . . 849
9.27.4.21 pack_sms_SLQSSmsGetMessageProtocol . . . 850
9.27.4.22 pack_sms_SLQSSmsSetRoutes . . . 850
9.27.4.23 pack_sms_SLQSSwiGetSMSStorage . . . 850
9.27.4.24 unpack_sms_GetSMSCAddress . . . 851
9.27.4.25 unpack_sms_SaveSMS . . . 851
9.27.4.26 unpack_sms_SendSMS . . . 851
9.27.4.27 unpack_sms_SetNewSMSCallback . . . 852
9.27.4.28 unpack_sms_SetNewSMSCallback_ind . . . 852
9.27.4.29 unpack_sms_SetSMSCAddress . . . 852
9.27.4.30 unpack_sms_SLQSDeleteSMS . . . 853
9.27.4.31 unpack_sms_SLQSGetIndicationRegister . . . 853
9.27.4.32 unpack_sms_SLQSGetMessageWaiting . . . 853
9.27.4.33 unpack_sms_SLQSGetSMS . . . 854
9.27.4.34 unpack_sms_SLQSGetSmsBroadcastConfig . . . 854
9.27.4.35 unpack_sms_SLQSGetSMSList . . . 854
9.27.4.36 unpack_sms_SLQSGetTransLayerInfo . . . 855
9.27.4.37 unpack_sms_SLQSGetTransNWRegInfo . . . 855
9.27.4.38 unpack_sms_SLQSModifySMSStatus . . . 855
9.27.4.39 unpack_sms_SLQSSendAsyncSMS . . . 856

9.27.4.40 unpack_sms_SLQSSetIndicationRegister . . . 856
9.27.4.41 unpack_sms_SLQSSetSmsBroadcastActivation . . . 856
9.27.4.42 unpack_sms_SLQSSetSmsBroadcastConfig . . . 857
9.27.4.43 unpack_sms_SLQSSetSmsStorage . . . 857
9.27.4.44 unpack_sms_SLQSSmsGetMaxStorageSize . . . 857
9.27.4.45 unpack_sms_SLQSSmsGetMessageProtocol . . . 858
9.27.4.46 unpack_sms_SLQSSmsSetRoutes . . . 858
9.27.4.47 unpack_sms_SLQSSwiGetSMSStorage . . . 858
9.27.4.48 unpack_sms_SLQSWmsMemoryFullCallBack_ind . . . 859
9.28 swiavms.h File Reference . . . 859
9.28.1 Macro Definition Documentation . . . 861
9.28.1.1 LITEQMI_MAX_SWIOMA_STR_LEN . . . 861
9.28.1.2 MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER . . . 861
9.28.1.3 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH . . . 861
9.28.1.4 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH . . . 861
9.28.1.5 MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH . . . 861
9.28.1.6 MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH . . . 861
9.28.1.7 MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH . . . 861
9.28.1.8 MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS . 862
9.28.1.9 MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH . . . 862
9.28.1.10 MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH . . . 862
9.28.2 Typedef Documentation . . . 862
9.28.2.1 PackSwiAvmsSetSettingsAPNInfo . . . 862
9.28.2.2 PackSwiAvmsSetSettingsConnectionRetryTimers . . . 862
9.28.2.3 PackSwiAvmsSetSettingsPeriodInfo . . . 862
9.28.2.4 PackSwiAVMSSettingsAPNInfo . . . 862
9.28.2.5 PackSwiAVMSSettingsConnectionRetryTimers . . . 862
9.28.2.6 PackSwiAVMSSettingsPeriodsInfo . . . 862
9.28.2.7 unpack_swiavms_SLQSAvMSEventReportInd_t . . . 862
9.28.2.8 unpack_swiavms_SLQSAvMSSessionGetInfo_t . . . 863
9.28.2.9 UnpackSwiAvmsEventReportBinaryUpdateSessionInfo . . . 863
9.28.2.10 UnpackSwiAvmsEventReportConfig . . . 864
9.28.2.11 UnpackSwiAvmsEventReportConnectionRequest . . . 864
9.28.2.12 UnpackSwiAvmsEventReportDataSessionStatus . . . 865
9.28.2.13 UnpackSwiAvmsEventReportHTTPStatus . . . 865
9.28.2.14 UnpackSwiAvmsEventReportNotification . . . 866
9.28.2.15 UnpackSwiAvmsEventReportPackageID . . . 867

9.28.2.16 UnpackSwiAvmsEventReportRegStatus . . . 867
9.28.2.17 UnpackSwiAvmsEventReportSessionType . . . 868
9.28.2.18 UnpackSwiAvmsEventReportWAMSParamChange . . . 868
9.28.3 Function Documentation . . . 868
9.28.3.1 pack_swiavms_SLQSAvMSGetSettings . . . 868
9.28.3.2 pack_swiavms_SLQSAvMSSendSelection . . . 869
9.28.3.3 pack_swiavms_SLQSAvMSSessionGetInfo . . . 869
9.28.3.4 pack_swiavms_SLQSAvmsSetEventReport . . . 870
9.28.3.5 pack_swiavms_SLQSAvMSSetSettings . . . 870
9.28.3.6 pack_swiavms_SLQSAvMSSetSettingsNoAutoRebootField . . . 871
9.28.3.7 pack_swiavms_SLQSAvMSStartSession . . . 872
9.28.3.8 pack_swiavms_SLQSAvMSStopSession . . . 872
9.28.3.9 unpack_swiavms_SLQSAvMSEventReportInd . . . 873
9.28.3.10 unpack_swiavms_SLQSAvMSGetSettings . . . 874
9.28.3.11 unpack_swiavms_SLQSAvMSSendSelection . . . 874
9.28.3.12 unpack_swiavms_SLQSAvMSSessionGetInfo . . . 874
9.28.3.13 unpack_swiavms_SLQSAvmsSetEventReport . . . 875
9.28.3.14 unpack_swiavms_SLQSAvMSSetSettings . . . 875
9.28.3.15 unpack_swiavms_SLQSAvMSStartSession . . . 876
9.28.3.16 unpack_swiavms_SLQSAvMSStopSession . . . 876
9.29 SwiDataTypes.h File Reference . . . 877
9.29.1 Detailed Description . . . 877
9.29.2 Macro Definition Documentation . . . 877
9.29.2.1 QMI_NO_LTE_FW_SUPPORT . . . 877
9.29.2.2 QMI_TLV_PLACEHOLDER . . . 877
9.29.2.3 SWI_API . . . 877
9.29.2.4 UNUSEDPARAM . . . 878
9.29.3 Typedef Documentation . . . 878
9.29.3.1 BOOL . . . 878
9.29.3.2 BYTE . . . 878
9.29.3.3 CHAR . . . 878
9.29.3.4 FLOAT . . . 878
9.29.3.5 INT32 . . . 878
9.29.3.6 INT8 . . . 878
9.29.3.7 LPCSTR . . . 878
9.29.3.8 SHORT . . . 878
9.29.3.9 ULONG . . . 878
9.29.3.10 ULONGLONG . . . 878
9.29.3.11 USHORT . . . 878
9.29.3.12 WORD . . . 878

9.30 swiloc.h File Reference . . . 878
9.30.1 Function Documentation . . . 878
9.30.1.1 pack_swiloc_SwiLocGetAutoStart . . . 878
9.30.1.2 pack_swiloc_SwiLocSetAutoStart . . . 879
9.30.1.3 unpack_swiloc_SwiLocGetAutoStart . . . 879
9.30.1.4 unpack_swiloc_SwiLocSetAutoStart . . . 879
9.31 swioma.h File Reference . . . 880
9.31.1 Macro Definition Documentation . . . 881
9.31.1.1 LITEQMI_MAX_SWIOMA_STR_LEN . . . 881
9.31.2 Function Documentation . . . 881
9.31.2.1 pack_swioma_SLQSOMADMAlertCallback . . . 881
9.31.2.2 pack_swioma_SLQSOMADMCancelSession . . . 881
9.31.2.3 pack_swioma_SLQSOMADMGetSessionInfo . . . 882
9.31.2.4 pack_swioma_SLQSOMADMGetSettings . . . 883
9.31.2.5 pack_swioma_SLQSOMADMSendSelection . . . 883
9.31.2.6 pack_swioma_SLQSOMADMSetSettings . . . 884
9.31.2.7 pack_swioma_SLQSOMADMStartSession . . . 885
9.31.2.8 unpack_swioma_SLQSOMADMAlertCallback . . . 885
9.31.2.9 unpack_swioma_SLQSOMADMAlertCallback_ind . . . 886
9.31.2.10 unpack_swioma_SLQSOMADMCancelSession . . . 886
9.31.2.11 unpack_swioma_SLQSOMADMGetSessionInfo . . . 887
9.31.2.12 unpack_swioma_SLQSOMADMGetSettings . . . 887
9.31.2.13 unpack_swioma_SLQSOMADMSendSelection . . . 887
9.31.2.14 unpack_swioma_SLQSOMADMSetSettings . . . 888
9.31.2.15 unpack_swioma_SLQSOMADMStartSession . . . 888
9.32 SWIWWANCMAPI.h File Reference . . . 889
9.33 uim.h File Reference . . . 889
9.33.1 Macro Definition Documentation . . . 892
9.33.1.1 MAX_DESCRIPTION_LENGTH . . . 892
9.33.1.2 MAX_ICCID_LENGTH . . . 892
9.33.1.3 MAX_NO_OF_APPLICATIONS . . . 892
9.33.1.4 MAX_NO_OF_SLOTS . . . 892
9.33.1.5 MAX_SLOTS_STATUS . . . 892
9.33.1.6 UIM_MAX_ACTIVE_PERS_FEATURES . . . 892
9.33.1.7 UIM_MAX_CONTENT_LENGTH . . . 892
9.33.1.8 UIM_MAX_DESCRIPTION_LENGTH . . . 892
9.33.1.9 UIM_MAX_NO_OF_APPLICATIONS . . . 892
9.33.1.10 UIM_MAX_NO_OF_SLOTS . . . 892
9.33.1.11 UIM_UINT8_MAX_STRING_SZ . . . 892
9.33.2 Function Documentation . . . 892

9.33.2.1 pack_uim_ChangePin . . . 892
9.33.2.2 pack_uim_GetCardStatus . . . 892
9.33.2.3 pack_uim_ReadTransparent . . . 893
9.33.2.4 pack_uim_SetPinProtection . . . 893
9.33.2.5 pack_uim_SLQSUIMAuthenticate . . . 893
9.33.2.6 pack_uim_SLQSUIMDepersonalization . . . 894
9.33.2.7 pack_uim_SLQSUIMEventRegister . . . 894
9.33.2.8 pack_uim_SLQSUIMGetConfiguration . . . 895
9.33.2.9 pack_uim_SLQSUIMGetFileAttributes . . . 895
9.33.2.10 pack_uim_SLQSUIMGetSlotsStatus . . . 895
9.33.2.11 pack_uim_SLQSUIMPowerDown . . . 896
9.33.2.12 pack_uim_SLQSUIMPowerUp . . . 896
9.33.2.13 pack_uim_SLQSUIMRefreshComplete . . . 896
9.33.2.14 pack_uim_SLQSUIMRefreshGetLastEvent . . . 897
9.33.2.15 pack_uim_SLQSUIMRefreshOK . . . 897
9.33.2.16 pack_uim_SLQSUIMRefreshRegister . . . 897
9.33.2.17 pack_uim_SLQSUIMReset . . . 898
9.33.2.18 pack_uim_SLQSUIMSwitchSlot . . . 898
9.33.2.19 pack_uim_UnblockPin . . . 899
9.33.2.20 pack_uim_VerifyPin . . . 899
9.33.2.21 unpack_uim_ChangePin . . . 899
9.33.2.22 unpack_uim_GetCardStatus . . . 900
9.33.2.23 unpack_uim_ReadTransparent . . . 900
9.33.2.24 unpack_uim_SetPinProtection . . . 900
9.33.2.25 unpack_uim_SetUimSlotStatusChangeCallback_ind . . . 901
9.33.2.26 unpack_uim_SLQSUIMAuthenticate . . . 901
9.33.2.27 unpack_uim_SLQSUIMDepersonalization . . . 901
9.33.2.28 unpack_uim_SLQSUIMEventRegister . . . 902
9.33.2.29 unpack_uim_SLQSUIMGetConfiguration . . . 902
9.33.2.30 unpack_uim_SLQSUIMGetFileAttributes . . . 903
9.33.2.31 unpack_uim_SLQSUIMGetSlotsStatus . . . 903
9.33.2.32 unpack_uim_SLQSUIMPowerDown . . . 903
9.33.2.33 unpack_uim_SLQSUIMPowerUp . . . 904
9.33.2.34 unpack_uim_SLQSUIMRefreshComplete . . . 904
9.33.2.35 unpack_uim_SLQSUIMRefreshGetLastEvent . . . 904
9.33.2.36 unpack_uim_SLQSUIMRefreshOK . . . 905
9.33.2.37 unpack_uim_SLQSUIMRefreshRegister . . . 905
9.33.2.38 unpack_uim_SLQSUIMReset . . . 905
9.33.2.39 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind . . . 905
9.33.2.40 unpack_uim_SLQSUIMSwitchSlot . . . 906

9.33.2.41 unpack_uim_UnblockPin . . . 906
9.33.2.42 unpack_uim_VerifyPin . . . 907
9.34 voice.h File Reference . . . 907
9.34.1 Macro Definition Documentation . . . 911
9.34.1.1 BARRING_PASSWORD_LENGTH . . . 911
9.34.1.2 MAX_VOICE_CALL_NO_LEN . . . 911
9.34.1.3 MAX_VOICE_DESCRIPTION_LENGTH . . . 911
9.34.1.4 MAXVOICEUSSDLENGTH . . . 911
9.34.1.5 VOICE_MAX_NO_OF_CALLS . . . 911
9.34.2 Function Documentation . . . 911
9.34.2.1 pack_voice_AnswerUSSD . . . 911
9.34.2.2 pack_voice_CancelUSSD . . . 912
9.34.2.3 pack_voice_OriginateUSSD . . . 912
9.34.2.4 pack_voice_SLQSOriginateUSSD . . . 912
9.34.2.5 pack_voice_SLQSVoiceALSSelectLine . . . 913
9.34.2.6 pack_voice_SLQSVoiceALSSetLineSwitching . . . 913
9.34.2.7 pack_voice_SLQSVoiceAnswerCall . . . 913
9.34.2.8 pack_voice_SLQSVoiceBindSubscription . . . 914
9.34.2.9 pack_voice_SLQSVoiceBurstDTMF . . . 914
9.34.2.10 pack_voice_SLQSVoiceDialCall . . . 914
9.34.2.11 pack_voice_SLQSVoiceEndCall . . . 915
9.34.2.12 pack_voice_SLQSVoiceGetAllCallInfo . . . 915
9.34.2.13 pack_voice_SLQSVoiceGetCallBarring . . . 916
9.34.2.14 pack_voice_SLQSVoiceGetCallForwardingStatus . . . 916
9.34.2.15 pack_voice_SLQSVoiceGetCallInfo . . . 916
9.34.2.16 pack_voice_SLQSVoiceGetCallWaiting . . . 917
9.34.2.17 pack_voice_SLQSVoiceGetCLIP . . . 917
9.34.2.18 pack_voice_SLQSVoiceGetCLIR . . . 917
9.34.2.19 pack_voice_SLQSVoiceGetCNAP . . . 918
9.34.2.20 pack_voice_SLQSVoiceGetCOLP . . . 918
9.34.2.21 pack_voice_SLQSVoiceGetCOLR . . . 918
9.34.2.22 pack_voice_SLQSVoiceGetConfig . . . 919
9.34.2.23 pack_voice_SLQSVoiceIndicationRegister . . . 919
9.34.2.24 pack_voice_SLQSVoiceManageCalls . . . 919
9.34.2.25 pack_voice_SLQSVoiceOrigUSSDNoWait . . . 920
9.34.2.26 pack_voice_SLQSVoiceSendFlash . . . 920
9.34.2.27 pack_voice_SLQSVoiceSetCallBarringPassword . . . 921
9.34.2.28 pack_voice_SLQSVoiceSetConfig . . . 921
9.34.2.29 pack_voice_SLQSVoiceSetPreferredPrivacy . . . 921
9.34.2.30 pack_voice_SLQSVoiceSetSUPSService . . . 922

9.34.2.31 pack_voice_SLQSVoiceStartContDTMF . . . 922
9.34.2.32 pack_voice_SLQSVoiceStopContDTMF . . . 922
9.34.2.33 unpack_voice_AnswerUSSD . . . 923
9.34.2.34 unpack_voice_CancelUSSD . . . 923
9.34.2.35 unpack_voice_OriginateUSSD . . . 923
9.34.2.36 unpack_voice_SLQSOriginateUSSD . . . 924
9.34.2.37 unpack_voice_SLQSVoiceALSSelectLine . . . 924
9.34.2.38 unpack_voice_SLQSVoiceALSSetLineSwitching . . . 924
9.34.2.39 unpack_voice_SLQSVoiceAnswerCall . . . 925
9.34.2.40 unpack_voice_SLQSVoiceBindSubscription . . . 925
9.34.2.41 unpack_voice_SLQSVoiceBurstDTMF . . . 925
9.34.2.42 unpack_voice_SLQSVoiceDialCall . . . 926
9.34.2.43 unpack_voice_SLQSVoiceEndCall . . . 926
9.34.2.44 unpack_voice_SLQSVoiceGetAllCallInfo . . . 927
9.34.2.45 unpack_voice_SLQSVoiceGetCallBarring . . . 927
9.34.2.46 unpack_voice_SLQSVoiceGetCallForwardingStatus . . . 927
9.34.2.47 unpack_voice_SLQSVoiceGetCallInfo . . . 928
9.34.2.48 unpack_voice_SLQSVoiceGetCallWaiting . . . 928
9.34.2.49 unpack_voice_SLQSVoiceGetCLIP . . . 928
9.34.2.50 unpack_voice_SLQSVoiceGetCLIR . . . 929
9.34.2.51 unpack_voice_SLQSVoiceGetCNAP . . . 929
9.34.2.52 unpack_voice_SLQSVoiceGetCOLP . . . 929
9.34.2.53 unpack_voice_SLQSVoiceGetCOLR . . . 930
9.34.2.54 unpack_voice_SLQSVoiceGetConfig . . . 930
9.34.2.55 unpack_voice_SLQSVoiceIndicationRegister . . . 930
9.34.2.56 unpack_voice_SLQSVoiceManageCalls . . . 931
9.34.2.57 unpack_voice_SLQSVoiceOrigUSSDNoWait . . . 931
9.34.2.58 unpack_voice_SLQSVoiceSendFlash . . . 931
9.34.2.59 unpack_voice_SLQSVoiceSetCallBarringPassword . . . 932
9.34.2.60 unpack_voice_SLQSVoiceSetConfig . . . 932
9.34.2.61 unpack_voice_SLQSVoiceSetPreferredPrivacy . . . 932
9.34.2.62 unpack_voice_SLQSVoiceSetSUPSService . . . 933
9.34.2.63 unpack_voice_SLQSVoiceStartContDTMF . . . 933
9.34.2.64 unpack_voice_SLQSVoiceStopContDTMF . . . 933
9.35 wds.h File Reference . . . 934
9.35.1 Macro Definition Documentation . . . 939
9.35.1.1 BYT_STAT_STAT_MASK . . . 939
9.35.1.2 IPV6_ADDRESS_ARRAY_SIZE . . . 939
9.35.1.3 MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE . . . 939
9.35.1.4 PACK_WDS_IPV4 . . . 939

9.35.1.5 PACK_WDS_IPV6 . . . 939
9.35.1.6 WDS_DHCP_MAX_NUM_OPTIONS . . . 939
9.35.1.7 WDS_DHCP_OPTION_DATA_BUF_SIZE . . . 939
9.35.2 Typedef Documentation . . . 939
9.35.2.1 UnpackQmiProfileInfo . . . 940
9.35.3 Function Documentation . . . 940
9.35.3.1 pack_wds_DHCPv4ClientLeaseChange . . . 940
9.35.3.2 pack_wds_GetAutoconnect . . . 940
9.35.3.3 pack_wds_GetByteTotals . . . 940
9.35.3.4 pack_wds_GetConnectionRate . . . 941
9.35.3.5 pack_wds_GetDataBearerTechnology . . . 941
9.35.3.6 pack_wds_GetDefaultProfile . . . 942
9.35.3.7 pack_wds_GetDefaultProfileNum . . . 942
9.35.3.8 pack_wds_GetDormancyState . . . 942
9.35.3.9 pack_wds_GetLastMobileIPError . . . 943
9.35.3.10 pack_wds_GetMobileIP . . . 943
9.35.3.11 pack_wds_GetMobileIPProfile . . . 944
9.35.3.12 pack_wds_GetPacketStatistics . . . 944
9.35.3.13 pack_wds_GetPacketStatus . . . 945
9.35.3.14 pack_wds_GetSessionDuration . . . 945
9.35.3.15 pack_wds_GetSessionState . . . 945
9.35.3.16 pack_wds_RMSetTransferStatistics . . . 946
9.35.3.17 pack_wds_SetAutoconnect . . . 946
9.35.3.18 pack_wds_SetDefaultProfile . . . 947
9.35.3.19 pack_wds_SetDefaultProfileNum . . . 947
9.35.3.20 pack_wds_SetMobileIP . . . 947
9.35.3.21 pack_wds_SetMobileIPParameters . . . 948
9.35.3.22 pack_wds_SetMobileIPProfile . . . 948
9.35.3.23 pack_wds_SLQSCreateProfile . . . 949
9.35.3.24 pack_wds_SLQSDeleteProfile . . . 949
9.35.3.25 pack_wds_SLQSGet3GPPConfigItem . . . 950
9.35.3.26 pack_wds_SLQSGetCurrDataSystemStat . . . 950
9.35.3.27 pack_wds_SLQSGetCurrentChannelRate . . . 950
9.35.3.28 pack_wds_SLQSGetDataBearerTechnology . . . 951
9.35.3.29 pack_wds_SLQSGetDUNCallInfo . . . 951
9.35.3.30 pack_wds_SLQSGetProfileSettings . . . 952
9.35.3.31 pack_wds_SLQSGetRuntimeSettings . . . 952
9.35.3.32 pack_wds_SLQSModifyProfile . . . 953
9.35.3.33 pack_wds_SLQSResetPacketStatics . . . 953
9.35.3.34 pack_wds_SLQSSet3GPPConfigItem . . . 953

9.35.3.35 pack_wds_SLQSSetIPFamilyPreference . . . 954
9.35.3.36 pack_wds_SLQSSetWdsEventCallback . . . 954
9.35.3.37 pack_wds_SLQSSGetDHCPv4ClientConfig . . . 955
9.35.3.38 pack_wds_SLQSSGetLoopback . . . 955
9.35.3.39 pack_wds_SLQSSSetDHCPv4ClientConfig . . . 955
9.35.3.40 pack_wds_SLQSSSetLoopback . . . 956
9.35.3.41 pack_wds_SLQSStartDataSession . . . 956
9.35.3.42 pack_wds_SLQSStopDataSession . . . 957
9.35.3.43 pack_wds_SLQSWdsGoActive . . . 957
9.35.3.44 pack_wds_SLQSWdsGoDormant . . . 957
9.35.3.45 pack_wds_SLQSWdsSetEventReport . . . 958
9.35.3.46 pack_wds_SLQSWdsSwiPDPRuntimeSettings . . . 958
9.35.3.47 unpack_wds_DHCPv4ClientLease_ind . . . 959
9.35.3.48 unpack_wds_DHCPv4ClientLeaseChange . . . 959
9.35.3.49 unpack_wds_GetAutoconnect . . . 959
9.35.3.50 unpack_wds_GetByteTotals . . . 960
9.35.3.51 unpack_wds_GetConnectionRate . . . 960
9.35.3.52 unpack_wds_GetDataBearerTechnology . . . 960
9.35.3.53 unpack_wds_GetDefaultProfile . . . 961
9.35.3.54 unpack_wds_GetDefaultProfileNum . . . 961
9.35.3.55 unpack_wds_GetDormancyState . . . 961
9.35.3.56 unpack_wds_GetLastMobileIPError . . . 962
9.35.3.57 unpack_wds_GetMobileIP . . . 962
9.35.3.58 unpack_wds_GetMobileIPProfile . . . 962
9.35.3.59 unpack_wds_GetPacketStatistics . . . 963
9.35.3.60 unpack_wds_GetPacketStatus . . . 963
9.35.3.61 unpack_wds_GetSessionDuration . . . 963
9.35.3.62 unpack_wds_GetSessionState . . . 964
9.35.3.63 unpack_wds_RMSetTransferStatistics . . . 964
9.35.3.64 unpack_wds_RMTransferStatistics_ind . . . 964
9.35.3.65 unpack_wds_SetAutoconnect . . . 965
9.35.3.66 unpack_wds_SetDefaultProfile . . . 965
9.35.3.67 unpack_wds_SetDefaultProfileNum . . . 965
9.35.3.68 unpack_wds_SetMobileIP . . . 966
9.35.3.69 unpack_wds_SetMobileIPParameters . . . 966
9.35.3.70 unpack_wds_SetMobileIPProfile . . . 966
9.35.3.71 unpack_wds_SLQSCreateProfile . . . 967
9.35.3.72 unpack_wds_SLQSDeleteProfile . . . 967
9.35.3.73 unpack_wds_SLQSGet3GPPConfigItem . . . 967
9.35.3.74 unpack_wds_SLQSGetCurrDataSystemStat . . . 968

9.35.3.75 unpack_wds_SLQSGetCurrentChannelRate . . . 968
9.35.3.76 unpack_wds_SLQSGetDataBearerTechnology . . . 968
9.35.3.77 unpack_wds_SLQSGetDUNCallInfo . . . 969
9.35.3.78 unpack_wds_SLQSGetProfileSettings . . . 969
9.35.3.79 unpack_wds_SLQSGetRuntimeSettings . . . 969
9.35.3.80 unpack_wds_SLQSModifyProfile . . . 970
9.35.3.81 unpack_wds_SLQSResetPacketStatics . . . 970
9.35.3.82 unpack_wds_SLQSSet3GPPConfigItem . . . 970
9.35.3.83 unpack_wds_SLQSSetIPFamilyPreference . . . 971
9.35.3.84 unpack_wds_SLQSSetPacketSrvStatusCallback . . . 971
9.35.3.85 unpack_wds_SLQSSetWdsEventCallback . . . 971
9.35.3.86 unpack_wds_SLQSSetWdsEventCallback_ind . . . 972
9.35.3.87 unpack_wds_SLQSSGetDHCPv4ClientConfig . . . 972
9.35.3.88 unpack_wds_SLQSSGetLoopback . . . 972
9.35.3.89 unpack_wds_SLQSSSetDHCPv4ClientConfig . . . 973
9.35.3.90 unpack_wds_SLQSSSetLoopback . . . 973
9.35.3.91 unpack_wds_SLQSStartDataSession . . . 973
9.35.3.92 unpack_wds_SLQSStopDataSession . . . 974
9.35.3.93 unpack_wds_SLQSWdsGoActive . . . 974
9.35.3.94 unpack_wds_SLQSWdsGoDormant . . . 974
9.35.3.95 unpack_wds_SLQSWdsSetEventReport . . . 975
9.35.3.96 unpack_wds_SLQSWdsSwiPDPRuntimeSettings . . . 975

Index **976**

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) 23
Streaming Download Protocol (lite-fw) 24

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Tables 25

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

_litefw_FirmwareInfo_ . . . 27
_PackSwiAvmsSetSettingsAPNInfo . . . 28
_PackSwiAvmsSetSettingsConnectionRetryTimers . . . 28
_PackSwiAvmsSetSettingsPeriodInfo . . . 29
_PackSwiAVMSSettingsAPNInfo . . . 29
_PackSwiAVMSSettingsConnectionRetryTimers . . . 29
_PackSwiAVMSSettingsPeriodsInfo . . . 30
_unpack_swiavms_SLQSAvMSEventReportInd_t . . . 30
_unpack_swiavms_SLQSAvMSSessionGetInfo_t . . . 31
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo . . . 32
_UnpackSwiAvmsEventReportConfig . . . 34
_UnpackSwiAvmsEventReportConnectionRequest . . . 35
_UnpackSwiAvmsEventReportDataSessionStatus . . . 35
_UnpackSwiAvmsEventReportHTTPStatus . . . 36
_UnpackSwiAvmsEventReportNotification . . . 38
_UnpackSwiAvmsEventReportPackageID . . . 39
_UnpackSwiAvmsEventReportRegStatus . . . 39
_UnpackSwiAvmsEventReportSessionType . . . 39
_UnpackSwiAvmsEventReportWAMSParamChange . . . 40
altSrcInfo_t . . . 40
appStats . . . 41
CarrierImage_t . . . 44
cdmaSSInfo . . . 46
connectionStatus . . . 46
crashInfoParams . . . 46
crashInformation . . . 47
currNetworkInfo . . . 48
dms_ActivationStatusTlv . . . 48
dms_OperatingModeTlv . . . 49
DMScustSettingInfo . . . 50
DMScustSettingList . . . 51
DMSgetCustomFeatureV2 . . . 51
DMSgetCustomInput . . . 52
dunchannelRate . . . 52
eriDataparams . . . 53
eTWSPLMNInfoTlv . . . 53
FMSImageElement . . . 54
FMSImageIdElement . . . 55

FMSImageIDEntries . . . 55
FMSImageList . . . 56
FMSPrefImageList . . . 57
hdrSSInfo . . . 57
image_info_t . . . 58
ipv6AddressInfo . . . 58
LibPackGPRSRequestedQoS . . . 59
LibpackProfile3GPP . . . 59
LibpackProfile3GPP2 . . . 65
LibPackprofile_3GPP . . . 70
LibPackprofile_3GPP2 . . . 76
LibPackQosClassID . . . 82
LibPackTFTIDParams . . . 82
LibPackUMTSQoS . . . 84
LibPackUMTSReqQoSSigInd . . . 87
loc_BdsSV . . . 87
loc_BdsSVInfo . . . 88
loc_CellDb . . . 88
loc_ClkInfo . . . 89
loc_GnssData . . . 90
loc_gpsTime . . . 92
loc_LocApplicationInfo . . . 92
loc_precisionDilution . . . 93
loc_satelliteInfo . . . 94
loc_sensorDataUsage . . . 96
loc_SV . . . 97
loc_SVInfo . . . 98
loc_svUsedforFix . . . 99
lteSSInfo . . . 99
messageModeTlv . . . 100
nas_acqOrderPref . . . 100
nas_ActPilotPNElement . . . 101
nas_AddCDMASysInfo . . . 101
nas_AddSysInfo . . . 102
nas_CallBarringSysInfo . . . 102
nas_callBarStatus . . . 103
nas_CDMAChannel . . . 104
nas_CDMAECIOThresh . . . 105
nas_CDMAInfo . . . 105
nas_CDMARSSIThresh . . . 106
nas_CDMASysInfo . . . 107
nas_CDMASysInfoExt . . . 111
nas_cellParams . . . 111
nas_CommInfo . . . 112
nas_CSGID . . . 114
nas_currentPLMN . . . 115
nas_dataSrvCapabilities . . . 116
nas_DataStatusDetail . . . 116
nas_detailSvcInfo . . . 119
nas_DeviceConfigDetail . . . 120
nas_dirNum . . . 121
nas_DRCParams . . . 121
nas_ecioListElement . . . 122
nas_ECIOThresh . . . 123
nas_errorRateListElement . . . 123
nas_GERANInfo . . . 124
nas_geranInstInfo . . . 126
nas_gsmCellInfo . . . 127

nas_GSMRSSIThresh . . . 128
nas_GSMSrvStatusInfo . . . 128
nas_GSMSysInfo . . . 129
nas_HDRECIOThresh . . . 132
nas_HDRIOThresh . . . 132
nas_HDRRSSIThresh . . . 133
nas_HDRSINRThresh . . . 134
nas_HDRSINRThreshold . . . 134
nas_HDRSysInfo . . . 135
nas_homeSIDNID . . . 137
nas_infoInterFreq . . . 138
nas_IOThresh . . . 139
nas_lteEARFCN . . . 139
nas_lteGsmCellInfo . . . 140
nas_LTEInfo . . . 141
nas_LTEInfoInterfreq . . . 144
nas_LTEInfoIntrafreq . . . 144
nas_LTEInfoNeighboringGSM . . . 146
nas_LTEInfoNeighboringWCDMA . . . 147
nas_ltePCI . . . 148
nas_lteRsrpinformation . . . 148
nas_LTERSRPThresh . . . 149
nas_LTERSRQThresh . . . 149
nas_LTERSSIThresh . . . 150
nas_LTESigRptCfg . . . 150
nas_LTESigRptConfig . . . 151
nas_lteSnrinformation . . . 152
nas_LTESNRThresh . . . 153
nas_LTESNRThreshold . . . 153
nas_LTESysInfo . . . 154
nas_lteWcdmaCellInfo . . . 156
nas_minBasedIMSI . . . 157
nas_MNRInfo . . . 158
nas_namName . . . 159
nas_netSelectionPref . . . 159
nas_NetworkStat1x . . . 160
nas_NetworkStatEVDO . . . 162
nas_nmrCellInfo . . . 164
nas_operatorNameString . . . 165
nas_OperatorPLMNData . . . 165
nas_operatorPLMNList . . . 166
nas_PhyCaAggPcellInfo . . . 167
nas_PhyCaAggScellDlBw . . . 168
nas_PhyCaAggScellIndex . . . 168
nas_PhyCaAggScellIndType . . . 169
nas_PhyCaAggScellInfo . . . 169
nas_PilotSetData . . . 172
nas_PilotSetParams . . . 172
nas_PLMNNetworkName . . . 173
nas_PLMNNetworkNameData . . . 174
nas_protocolSubtypeElement . . . 175
nas_qaQmi3Gpp2TimeZone . . . 177
nas_QmiNas3GppNetworkInfo . . . 177
nas_QmiNas3GppNetworkRAT . . . 178
nas_QmisNasPcsDigit . . . 179
nas_RejectReasonTlv . . . 180
nas_RFInfoTlv . . . 180
nas_roamIndList . . . 181

nas_RSRPThresh . . . 181
nas_rsrqInformation . . . 182
nas_RSRQThresh . . . 183
nas_RSSIThresh . . . 183
nas_rxInfo . . . 184
nas_RxSigInfo . . . 185
nas_rxSignalStrengthListElement . . . 186
nas_SccRxInfo . . . 187
nas_serviceProviderName . . . 188
nas_servSystem . . . 188
nas_sidNid . . . 190
nas_SignalStrengthTlv . . . 190
nas_SLQSSignalStrengthsIndReq . . . 191
nas_SLQSSignalStrengthsInformation . . . 192
nas_SLQSSignalStrengthsTlv . . . 193
nas_SrvStatusInfo . . . 193
nas_sysInfoCommon . . . 194
nas_TDSCDMAECIOThresh . . . 196
nas_TDSCDMARSCPThresh . . . 197
nas_TDSCDMARSSIThresh . . . 197
nas_TDSCDMASINRCONFThresh . . . 198
nas_TDSCDMASINRThresh . . . 198
nas_timeInfo . . . 199
nas_trueIMSI . . . 200
nas_txInfo . . . 201
nas_UMTSInfo . . . 202
nas_UMTSinstInfo . . . 204
nas_umtsLTENbrCell . . . 204
nas_UniversalTime . . . 205
nas_wcdmaCellInfo . . . 206
nas_WCDMAECIOThresh . . . 207
nas_WCDMAInfoLTENeighborCell . . . 208
nas_WCDMARSSIThresh . . . 209
nas_WCDMASysInfo . . . 209
nas_wcdmaUARFCN . . . 213
NASBandPreferenceTlv . . . 213
NASEmergencyModeTlv . . . 213
NasGetLTECphyCaInfo . . . 214
NASGWAcqOrderPrefTlv . . . 214
NASLTEBandPreferenceTlv . . . 214
NASLteNasReleaseInfoTlv . . . 215
NASModePreferenceTlv . . . 215
NASNetSelPreferenceTlv . . . 215
NASOTAMessageTlv . . . 215
NASPhyCaAggPcellInfo . . . 216
NASPhyCaAggScellDlBw . . . 217
NASPhyCaAggScellIndex . . . 217
NASPhyCaAggScellIndType . . . 218
NASPhyCaAggScellInfo . . . 219
NASPRLPreferenceTlv . . . 220
NASQmiCbkNasSwiOTAMessageInd . . . 220
NASQmiCbkNasSystemSelPrefInd . . . 220
NASRoamPreferenceTlv . . . 221
NASServDomainPrefTlv . . . 221
NASServingSystemInfo . . . 221
NASTimeInfoTlv . . . 223
newMTMessageTlv . . . 223
pack_dms_ActivateAutomatic_t . . . 223

pack_dms_GetCustFeaturesV2_t . . . 224
pack_dms_ResetToFactoryDefaults_t . . . 224
pack_dms_SetActivationStatusCallback_t . . . 225
pack_dms_SetCrashAction_t . . . 225
pack_dms_SetCustFeature_t . . . 226
pack_dms_SetCustFeaturesV2_t . . . 226
pack_dms_SetEventReport_t . . . 227
pack_dms_SetPower_t . . . 227
pack_dms_SetUSBComp_t . . . 228
pack_dms_SLQSDmsSwiIndicationRegister_t . . . 228
pack_dms_SLQSSwiGetCrashInfo_t . . . 228
pack_dms_SLQSSwiSetDyingGaspCfg_t . . . 229
pack_dms_SLQSSwiSetHostDevInfo_t . . . 229
pack_dms_SLQSSwiSetOSInfo_t . . . 230
pack_dms_UIMChangePIN_t . . . 230
pack_dms_UIMGetControlKeyStatus_t . . . 231
pack_dms_UIMGetICCID_t . . . 232
pack_dms_UIMSetControlKeyProtection_t . . . 232
pack_dms_UIMSetPINProtection_t . . . 233
pack_dms_UIMUnblockControlKey_t . . . 233
pack_dms_UIMUnblockPIN_t . . . 234
pack_dms_UIMVerifyPIN_t . . . 235
pack_fms_GetImagesPreference_t . . . 235
pack_fms_GetStoredImages_t . . . 236
pack_fms_SetImagesPreference_t . . . 236
pack_loc_Delete_Assist_Data_t . . . 237
pack_loc_EventRegister_t . . . 238
pack_loc_SetExtPowerState_t . . . 240
pack_loc_SetOperationMode_t . . . 240
pack_loc_SLQSLOCGetBestAvailPos_t . . . 241
pack_loc_SLQSLOCInjectPosition_t . . . 241
pack_loc_SLQSLOCInjectSensorData_t . . . 246
pack_loc_SLQSLOCInjectUTCTime_t . . . 248
pack_loc_SLQSLOCSetCradleMountConfig_t . . . 249
pack_loc_Start_t . . . 249
pack_loc_Stop_t . . . 251
pack_nas_InitiateDomainAttach_t . . . 251
pack_nas_SetACCOLC_t . . . 252
pack_nas_SetCDMANetworkParameters_t . . . 252
pack_nas_SetNetworkPreference_t . . . 254
pack_nas_SLQSConfigSigInfo_t . . . 255
pack_nas_SLQSGetPLMNName_t . . . 256
pack_nas_SLQSInitiateNetworkRegistration_t . . . 257
pack_nas_SLQSNasConfigSigInfo2_t . . . 258
pack_nas_SLQSNasGet3GPP2Subscription_t . . . 262
pack_nas_SLQSNasGetTxRxInfo_t . . . 263
pack_nas_SLQSNasIndicationRegisterExt_t . . . 263
pack_nas_SLQSNasSwiIndicationRegister_t . . . 266
pack_nas_SLQSNASSwiSetChannelLock_t . . . 267
pack_nas_SLQSSetBandPreference_t . . . 268
pack_nas_SLQSSetSignalStrengthsCallback_t . . . 270
pack_nas_SLQSSetSysSelectionPref_t . . . 270
pack_nas_SLQSSwiPSDetach_t . . . 275
pack_pds_SetPDSDefaults_t . . . 275
pack_pds_SetPDSState_t . . . 276
pack_pds_SetPortAutomaticTracking_t . . . 276
pack_qmi_t . . . 277
pack_qos_SLQSQosSwiReadApnExtraParams_t . . . 277

pack_qos_SLQSQosSwiReadDataStats_t . . . 278
pack_qos_SLQSSetQosEventCallback_t . . . 278
pack_rms_SetSMSWake_t . . . 279
pack_sar_SLQSSetRfSarState_t . . . 279
pack_sms_SaveSMS_t . . . 280
pack_sms_SendSMS_t . . . 281
pack_sms_SetNewSMSCallback_t . . . 281
pack_sms_SetSMSCAddress_t . . . 282
pack_sms_SLQSDeleteSMS_t . . . 282
pack_sms_SLQSGetSMS_t . . . 283
pack_sms_SLQSGetSmsBroadcastConfig_t . . . 284
pack_sms_SLQSGetSMSList_t . . . 284
pack_sms_SLQSModifySMSStatus_t . . . 285
pack_sms_SLQSSendAsyncSMS_t . . . 286
pack_sms_SLQSSetIndicationRegister_t . . . 286
pack_sms_SLQSSetSmsBroadcastActivation_t . . . 287
pack_sms_SLQSSetSmsBroadcastConfig_t . . . 287
pack_sms_SLQSSetSmsStorage_t . . . 288
pack_sms_SLQSSmsGetMaxStorageSize_t . . . 288
pack_sms_SLQSSmsSetRoutes_t . . . 289
pack_swiavms_SLQSAvMSSendSelection_t . . . 289
pack_swiavms_SLQSAvMSSetSettings_t . . . 290
pack_swiavms_SLQSAvMSStartSession_t . . . 292
pack_swiavms_SLQSAvMSStopSession_t . . . 292
pack_swiloc_SwiLocSetAutoStart_t . . . 293
pack_swioma_SLQSOMADMCancelSession_t . . . 294
pack_swioma_SLQSOMADMGetSessionInfo_t . . . 295
pack_swioma_SLQSOMADMSendSelection_t . . . 295
pack_swioma_SLQSOMADMSetSettings_t . . . 296
pack_swioma_SLQSOMADMStartSession_t . . . 297
pack_uim_ChangePin_t . . . 297
pack_uim_ReadTransparent_t . . . 298
pack_uim_SetPinProtection_t . . . 300
pack_uim_SLQSUIMAuthenticate_t . . . 301
pack_uim_SLQSUIMDepersonalization_t . . . 301
pack_uim_SLQSUIMEventRegister_t . . . 302
pack_uim_SLQSUIMGetConfiguration_t . . . 302
pack_uim_SLQSUIMGetFileAttributes_t . . . 303
pack_uim_SLQSUIMPowerDown_t . . . 303
pack_uim_SLQSUIMPowerUp_t . . . 304
pack_uim_SLQSUIMRefreshComplete_t . . . 304
pack_uim_SLQSUIMRefreshGetLastEvent_t . . . 305
pack_uim_SLQSUIMRefreshOK_t . . . 305
pack_uim_SLQSUIMRefreshRegister_t . . . 306
pack_uim_SLQSUIMSwitchSlot_t . . . 306
pack_uim_UnblockPin_t . . . 307
pack_uim_VerifyPin_t . . . 308
pack_voice_AnswerUSSD_t . . . 309
pack_voice_OriginateUSSD_t . . . 310
pack_voice_SLQSOriginateUSSD_t . . . 310
pack_voice_SLQSVoiceALSSelectLine_t . . . 311
pack_voice_SLQSVoiceALSSetLineSwitching_t . . . 311
pack_voice_SLQSVoiceAnswerCall_t . . . 311
pack_voice_SLQSVoiceBindSubscription_t . . . 312
pack_voice_SLQSVoiceBurstDTMF_t . . . 312
pack_voice_SLQSVoiceDialCall_t . . . 313
pack_voice_SLQSVoiceEndCall_t . . . 315
pack_voice_SLQSVoiceGetCallBarring_t . . . 315

pack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 316
pack_voice_SLQSVoiceGetCallInfo_t . . . 316
pack_voice_SLQSVoiceGetCallWaiting_t . . . 317
pack_voice_SLQSVoiceGetConfig_t . . . 317
pack_voice_SLQSVoiceIndicationRegister_t . . . 319
pack_voice_SLQSVoiceManageCalls_t . . . 320
pack_voice_SLQSVoiceOrigUSSDNoWait_t . . . 321
pack_voice_SLQSVoiceSendFlash_t . . . 321
pack_voice_SLQSVoiceSetCallBarringPassword_t . . . 322
pack_voice_SLQSVoiceSetConfig_t . . . 323
pack_voice_SLQSVoiceSetPreferredPrivacy_t . . . 324
pack_voice_SLQSVoiceSetSUPSService_t . . . 324
pack_voice_SLQSVoiceStartContDTMF_t . . . 327
pack_voice_SLQSVoiceStopContDTMF_t . . . 328
pack_wds_DHCPv4ClientLeaseChange_t . . . 328
pack_wds_GetDefaultProfile_t . . . 328
pack_wds_GetDefaultProfileNum_t . . . 329
pack_wds_GetDormancyState_t . . . 329
pack_wds_GetLastMobileIPError_t . . . 329
pack_wds_GetMobileIP_t . . . 329
pack_wds_GetMobileIPProfile_t . . . 329
pack_wds_GetPacketStatistics_t . . . 330
pack_wds_GetPacketStatus_t . . . 330
pack_wds_GetSessionDuration_t . . . 330
pack_wds_RMSetTransferStatistics_t . . . 330
pack_wds_SetAutoconnect_t . . . 331
pack_wds_SetDefaultProfile_t . . . 331
pack_wds_SetDefaultProfileNum_t . . . 332
pack_wds_SetMobileIP_t . . . 332
pack_wds_SetMobileIPParameters_t . . . 333
pack_wds_SetMobileIPProfile_t . . . 334
pack_wds_SLQSCreateProfile_t . . . 335
pack_wds_SLQSDeleteProfile_t . . . 336
pack_wds_SLQSGetCurrDataSystemStat_t . . . 336
pack_wds_SLQSGetDataBearerTechnology_t . . . 337
pack_wds_SLQSGetDUNCallInfo_t . . . 337
pack_wds_SLQSGetProfileSettings_t . . . 337
pack_wds_SLQSGetRuntimeSettings_t . . . 338
pack_wds_SLQSModifyProfile_t . . . 339
pack_wds_SLQSSet3GPPConfigItem_t . . . 340
pack_wds_SLQSSetIPFamilyPreference_t . . . 342
pack_wds_SLQSSetWdsEventCallback_t . . . 342
pack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 343
pack_wds_SLQSSSetDHCPv4ClientConfig_t . . . 343
pack_wds_SLQSSSetLoopback_t . . . 344
pack_wds_SLQSStartDataSession_t . . . 344
pack_wds_SLQSStopDataSession_t . . . 345
pack_wds_SLQSWdsSetEventReport_t . . . 346
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 348
PackCreateProfileOut . . . 348
packgetDyingGaspCfg . . . 348
packgetDyingGaspStatistics . . . 349
qmiSmsMessageList . . . 349
qmiWDSDataBearerTechnology . . . 350
RFBandInfoElements . . . 350
rmTrasnferStaticsReq . . . 351
sensorData_t . . . 351
slot_t . . . 352

slotInf . . . 353
slots_t . . . 355
sms_BroadcastConfig . . . 355
sms_CDMABroadcastConfig . . . 356
sms_getIndicationReg . . . 356
sms_getMsgWaitingInfo . . . 357
sms_getTransLayerInfo . . . 358
sms_getTransNWRegInfo . . . 359
sms_maxStorageSizeReq . . . 359
sms_maxStorageSizeResp . . . 360
sms_messageWaitingInfoContent . . . 360
sms_msgProtocolResp . . . 361
sms_qaQmi3GPP2BroadcastCfgInfo . . . 361
sms_qaQmi3GPPBroadcastCfgInfo . . . 362
sms_routeEntry . . . 363
sms_sendAsyncsmsParams . . . 364
sms_setIndicationReg . . . 366
sms_setRoutesReq . . . 367
sms_transLayerInfo . . . 368
sMSCAddress . . . 369
sMSCAddressTlv . . . 369
sMSEtwsMessage . . . 369
sMSEtwsMessageTlv . . . 370
sMSEtwsPlmn . . . 371
sMSMessageMode . . . 371
sMSMTMessage . . . 371
sMSOnIMS . . . 372
sMSOnIMSTlv . . . 372
sMSTransferRouteMTMessage . . . 373
tdscdmaSigInfoExt . . . 373
tempData_t . . . 374
transferRouteMessageTlv . . . 375
transferStatInd . . . 375
uim_appStatus . . . 376
uim_authenticateResult . . . 379
uim_authenticationData . . . 379
uim_cardResult . . . 380
uim_cardStatus . . . 381
uim_changeUIMPIN . . . 382
uim_depersonalizationInformation . . . 383
uim_encryptedPIN1 . . . 384
uim_fileAttributes . . . 385
uim_fileInfo . . . 388
uim_hotSwapStatus . . . 389
uim_personalizationStatus . . . 389
uim_readResult . . . 390
uim_readTransparentInfo . . . 391
uim_refreshevent . . . 391
uim_registerRefresh . . . 393
uim_remainingRetries . . . 394
uim_sessionInformation . . . 395
uim_setPINProtection . . . 395
uim_slotInfo . . . 396
uim_UIMSessionInformation . . . 398
uim_unblockUIMPIN . . . 399
uim_verifyUIMPIN . . . 400
unpack_dms_GetActivationState_t . . . 400
unpack_dms_GetBandCapability_t . . . 401

unpack_dms_GetCrashAction_t . . . 401
unpack_dms_GetCustFeature_t . . . 401
unpack_dms_GetCustFeaturesV2_t . . . 402
unpack_dms_GetDeviceCap_t . . . 403
unpack_dms_GetDeviceCapabilities_t . . . 403
unpack_dms_GetDeviceHardwareRev_t . . . 404
unpack_dms_GetDeviceMfr_t . . . 404
unpack_dms_GetDeviceSerialNumbers_t . . . 404
unpack_dms_GetFirmwareInfo_t . . . 405
unpack_dms_GetFirmwareRevision_t . . . 406
unpack_dms_GetFirmwareRevisions_t . . . 406
unpack_dms_GetFSN_t . . . 407
unpack_dms_GetHardwareRevision_t . . . 407
unpack_dms_GetIMSI_t . . . 408
unpack_dms_GetManufacturer_t . . . 408
unpack_dms_GetModelID_t . . . 408
unpack_dms_GetNetworkTime_t . . . 409
unpack_dms_GetOfflineReason_t . . . 410
unpack_dms_GetPower_t . . . 410
unpack_dms_GetPRLVersion_t . . . 411
unpack_dms_GetSerialNumbers_t . . . 411
unpack_dms_GetUSBComp_t . . . 412
unpack_dms_GetVoiceNumber_t . . . 412
unpack_dms_ResetToFactoryDefaults_t . . . 412
unpack_dms_SetActivationStatusCallback_t . . . 413
unpack_dms_SetCrashAction_t . . . 413
unpack_dms_SetCustFeature_t . . . 414
unpack_dms_SetCustFeaturesV2_t . . . 414
unpack_dms_SetEventReport_ind_t . . . 414
unpack_dms_SetEventReport_t . . . 415
unpack_dms_SetFirmwarePreference_t . . . 415
unpack_dms_SetPower_t . . . 415
unpack_dms_SetUSBComp_t . . . 415
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t . . . 416
unpack_dms_SLQSDmsSwiGetResetInfo_t . . . 417
unpack_dms_SLQSDmsSwiIndicationRegister_t . . . 417
unpack_dms_SLQSGetBandCapability_t . . . 418
unpack_dms_SLQSGetERIFile_t . . . 421
unpack_dms_SLQSSwiClearDyingGaspStatistics_t . . . 421
unpack_dms_SLQSSwiGetCrashInfo_t . . . 422
unpack_dms_SLQSSwiGetDyingGaspCfg_t . . . 422
unpack_dms_SLQSSwiGetDyingGaspStatistics_t . . . 423
unpack_dms_SLQSSwiGetFirmwareCurr_t . . . 423
unpack_dms_SLQSSwiGetFwUpdateStatus_t . . . 424
unpack_dms_SLQSSwiGetHostDevInfo_t . . . 426
unpack_dms_SLQSSwiGetOSInfo_t . . . 426
unpack_dms_SLQSSwiGetSerialNoExt_t . . . 427
unpack_dms_SLQSSwiSetDyingGaspCfg_t . . . 427
unpack_dms_SLQSSwiSetHostDevInfo_t . . . 428
unpack_dms_SLQSSwiSetOSInfo_t . . . 428
unpack_dms_SLQSUIMGetState_t . . . 429
unpack_dms_UIMGetControlKeyStatus_t . . . 429
unpack_dms_UIMGetICCID_t . . . 430
unpack_dms_UIMGetPINStatus_t . . . 431
unpack_dms_UIMSetControlKeyProtection_t . . . 432
unpack_dms_UIMSetPINProtection_t . . . 433
unpack_dms_UIMUnblockControlKey_t . . . 433
unpack_fms_GetImagesPreference_t . . . 434

unpack_fms_GetStoredImages_t . . . 435
unpack_fms_SetImagesPreference_t . . . 435
unpack_loc_BestAvailPos_Ind_t . . . 436
unpack_loc_Delete_Assist_Data_t . . . 441
unpack_loc_DeleteAssistData_Ind_t . . . 442
unpack_loc_EngineState_Ind_t . . . 443
unpack_loc_EventRegister_t . . . 443
unpack_loc_GnssSvInfo_Ind_t . . . 444
unpack_loc_PositionRpt_Ind_t . . . 444
unpack_loc_SetExtPowerConfig_Ind_t . . . 451
unpack_loc_SetExtPowerState_t . . . 451
unpack_loc_SetOperationMode_Ind_t . . . 452
unpack_loc_SetOperationMode_t . . . 452
unpack_loc_SLQSLOCGetBestAvailPos_t . . . 453
unpack_loc_Start_t . . . 453
unpack_loc_Stop_t . . . 453
unpack_nas_GetCDMANetworkParameters_t . . . 454
unpack_nas_GetHomeNetwork3GPP2_t . . . 455
unpack_nas_GetHomeNetwork_t . . . 457
unpack_nas_GetNetworkPreference_t . . . 457
unpack_nas_GetRFInfo_t . . . 458
unpack_nas_GetServingNetwork_t . . . 459
unpack_nas_GetServingNetworkCapabilities_t . . . 460
unpack_nas_GetSignalStrengths_t . . . 460
unpack_nas_PerformNetworkScan_t . . . 461
unpack_nas_SetDataCapabilitiesCallback_ind_t . . . 462
unpack_nas_SetEventReportInd_t . . . 462
unpack_nas_SetNasLTECphyCaIndCallback_ind_t . . . 462
unpack_nas_SetNetworkPreference_t . . . 463
unpack_nas_SetRoamingIndicatorCallback_ind_t . . . 464
unpack_nas_SetServingSystemCallback_ind_t . . . 465
unpack_nas_SLQSGetErrorRate_t . . . 465
unpack_nas_SlqsGetLTECphyCAInfo_t . . . 466
unpack_nas_SLQSGetNetworkTime_t . . . 467
unpack_nas_SLQSGetOperatorNameData_t . . . 467
unpack_nas_SLQSGetPLMNName_t . . . 468
unpack_nas_SLQSGetServingSystem_t . . . 469
unpack_nas_SLQSGetSignalStrength_t . . . 471
unpack_nas_SLQSGetSysInfo_t . . . 472
unpack_nas_SLQSGetSysSelectionPref_t . . . 475
unpack_nas_SLQSNasGet3GPP2Subscription_t . . . 478
unpack_nas_SLQSNasGetCellLocationInfo_t . . . 479
unpack_nas_SLQSNasGetHDRColorCode_t . . . 481
unpack_nas_SLQSNasGetSigInfo_t . . . 481
unpack_nas_SLQSNasGetTxRxInfo_t . . . 482
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t . . . 482
unpack_nas_SLQSNasSigInfoCallback_ind_t . . . 483
unpack_nas_SLQSNASSwiGetChannelLock_t . . . 484
unpack_nas_SLQSNasSwiModemStatus_t . . . 485
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t . . . 485
unpack_nas_SLQSNasTimerCallback_ind_t . . . 486
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t . . . 486
unpack_nas_SLQSSwiGetHDRPersonality_t . . . 487
unpack_nas_SLQSSwiGetHDRProtSubtype_t . . . 487
unpack_nas_SLQSSwiGetHRPDStats_t . . . 488
unpack_nas_SLQSSwiGetLteCQI_t . . . 489
unpack_nas_SLQSSwiGetLteSccRxInfo_t . . . 489
unpack_nas_SLQSSwiNetworkDebug_t . . . 490

unpack_nas_SLQSSysInfoCallback_ind_t . . . 491
unpack_omaDmConfigTlv_t . . . 493
unpack_omaDmFotaTlv_t . . . 494
unpack_omaDmNotificationsTlv_t . . . 496
unpack_pds_GetPDSDefaults_t . . . 497
unpack_pds_GetPDSState_t . . . 497
unpack_pds_GetPortAutomaticTracking_t . . . 498
unpack_qmi_t . . . 498
unpack_qos_dataRate_t . . . 499
unpack_qos_IPv4Addr_t . . . 499
unpack_qos_IPv6Addr_t . . . 500
unpack_qos_IPv6TrafCls_t . . . 500
unpack_qos_pktErrRate_t . . . 501
unpack_qos_Port_t . . . 501
unpack_qos_QosFlowInfo_t . . . 502
unpack_qos_QosFlowInfoState_t . . . 503
unpack_qos_SLQSQosGetNetworkStatus_t . . . 504
unpack_qos_SLQSQosSwiReadApnExtraParams_t . . . 504
unpack_qos_SLQSQosSwiReadDataStats_t . . . 506
unpack_qos_SLQSSetQosEventCallback_ind_t . . . 507
unpack_qos_SLQSSetQosNWStatusCallback_ind_t . . . 507
unpack_qos_SLQSSetQosPriEventCallback_ind_t . . . 508
unpack_qos_SLQSSetQosStatusCallback_ind_t . . . 508
unpack_qos_swiQosFilter_t . . . 510
unpack_qos_swiQosFlow_t . . . 513
unpack_qos_tokenBucket_t . . . 517
unpack_qos_Tos_t . . . 518
unpack_QosFlowStat_t . . . 518
unpack_rms_GetSMSWake_t . . . 519
unpack_RMTransferStatistics_ind_t . . . 519
unpack_sar_SLQSGetRfSarState_t . . . 520
unpack_sms_GetSMSCAddress_t . . . 521
unpack_sms_SaveSMS_t . . . 522
unpack_sms_SendSMS_t . . . 522
unpack_sms_SetNewSMSCallback_ind_t . . . 522
unpack_sms_SetNewSMSCallback_t . . . 524
unpack_sms_SLQSDeleteSMS_t . . . 524
unpack_sms_SLQSGetIndicationRegister_t . . . 524
unpack_sms_SLQSGetMessageWaiting_t . . . 524
unpack_sms_SLQSGetSMS_t . . . 525
unpack_sms_SLQSGetSmsBroadcastConfig_t . . . 525
unpack_sms_SLQSGetSMSList_t . . . 526
unpack_sms_SLQSGetTransLayerInfo_t . . . 526
unpack_sms_SLQSGetTransNWRegInfo_t . . . 527
unpack_sms_SLQSModifySMSStatus_t . . . 527
unpack_sms_SLQSSmsGetMaxStorageSize_t . . . 527
unpack_sms_SLQSSmsGetMessageProtocol_t . . . 528
unpack_sms_SLQSSwiGetSMSStorage_t . . . 528
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t . . . 529
unpack_swiavms_SLQSAvMSGetSettings_t . . . 529
unpack_swiavms_SLQSAvMSSendSelection_t . . . 530
unpack_swiavms_SLQSAvmsSetEventReport_t . . . 530
unpack_swiavms_SLQSAvMSSetSettings_t . . . 531
unpack_swiavms_SLQSAvMSStartSession_t . . . 531
unpack_swiavms_SLQSAvMSStopSession_t . . . 532
unpack_swiloc_SwiLocGetAutoStart_t . . . 532
unpack_swioma_SLQSOMADMAlertCallback_ind_t . . . 534
unpack_swioma_SLQSOMADMGetSessionInfo_t . . . 534

unpack_swioma_SLQSOMADMGetSettings_t . . . 537
unpack_swioma_SLQSOMADMStartSession_t . . . 538
unpack_uim_ChangePin_t . . . 539
unpack_uim_GetCardStatus_t . . . 540
unpack_uim_ReadTransparent_t . . . 540
unpack_uim_SetPinProtection_t . . . 541
unpack_uim_SetUimSlotStatusChangeCallback_ind_t . . . 542
unpack_uim_SLQSUIMAuthenticate_t . . . 542
unpack_uim_SLQSUIMDepersonalization_t . . . 543
unpack_uim_SLQSUIMEventRegister_t . . . 544
unpack_uim_SLQSUIMGetConfiguration_t . . . 544
unpack_uim_SLQSUIMGetFileAttributes_t . . . 545
unpack_uim_SLQSUIMGetSlotsStatus_t . . . 545
unpack_uim_SLQSUIMRefreshGetLastEvent_t . . . 546
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t . . . 546
unpack_uim_UnblockPin_t . . . 547
unpack_uim_VerifyPin_t . . . 547
unpack_voice_SLQSOriginateUSSD_t . . . 548
unpack_voice_SLQSVoiceAnswerCall_t . . . 549
unpack_voice_SLQSVoiceBurstDTMF_t . . . 549
unpack_voice_SLQSVoiceDialCall_t . . . 549
unpack_voice_SLQSVoiceEndCall_t . . . 550
unpack_voice_SLQSVoiceGetAllCallInfo_t . . . 551
unpack_voice_SLQSVoiceGetCallBarring_t . . . 553
unpack_voice_SLQSVoiceGetCallForwardingStatus_t . . . 555
unpack_voice_SLQSVoiceGetCallInfo_t . . . 556
unpack_voice_SLQSVoiceGetCallWaiting_t . . . 558
unpack_voice_SLQSVoiceGetCLIP_t . . . 560
unpack_voice_SLQSVoiceGetCLIR_t . . . 561
unpack_voice_SLQSVoiceGetCNAP_t . . . 562
unpack_voice_SLQSVoiceGetCOLP_t . . . 563
unpack_voice_SLQSVoiceGetCOLR_t . . . 565
unpack_voice_SLQSVoiceGetConfig_t . . . 566
unpack_voice_SLQSVoiceManageCalls_t . . . 568
unpack_voice_SLQSVoiceSendFlash_t . . . 568
unpack_voice_SLQSVoiceSetCallBarringPassword_t . . . 569
unpack_voice_SLQSVoiceSetConfig_t . . . 570
unpack_voice_SLQSVoiceSetSUPSService_t . . . 571
unpack_voice_SLQSVoiceStartContDTMF_t . . . 572
unpack_voice_SLQSVoiceStopContDTMF_t . . . 573
unpack_wds_DHCPv4ClientLease_ind_t . . . 573
unpack_wds_GetAutoconnect_t . . . 574
unpack_wds_GetByteTotals_t . . . 574
unpack_wds_GetConnectionRate_t . . . 575
unpack_wds_GetDataBearerTechnology_t . . . 575
unpack_wds_GetDefaultProfile_t . . . 576
unpack_wds_GetDefaultProfileNum_t . . . 577
unpack_wds_GetDormancyState_t . . . 578
unpack_wds_GetLastMobileIPError_t . . . 578
unpack_wds_GetMobileIP_t . . . 578
unpack_wds_GetMobileIPProfile_t . . . 579
unpack_wds_GetPacketStatistics_t . . . 580
unpack_wds_GetPacketStatus_t . . . 581
unpack_wds_GetSessionDuration_t . . . 582
unpack_wds_GetSessionState_t . . . 583
unpack_wds_RMSetTransferStatistics_t . . . 583
unpack_wds_SetMobileIPProfile_t . . . 583
unpack_wds_SLQSCreateProfile_t . . . 583

unpack_wds_SLQSDeleteProfile_t . . . 584
unpack_wds_SLQSGet3GPPConfigItem_t . . . 584
unpack_wds_SLQSGetCurrDataSystemStat_t . . . 585
unpack_wds_SLQSGetCurrentChannelRate_t . . . 586
unpack_wds_SLQSGetDataBearerTechnology_t . . . 587
unpack_wds_SLQSGetDUNCallInfo_t . . . 587
unpack_wds_SLQSGetProfileSettings_t . . . 588
unpack_wds_SLQSGetRuntimeSettings_t . . . 589
unpack_wds_SLQSModifyProfile_t . . . 591
unpack_wds_SLQSSetIPFamilyPreference_t . . . 591
unpack_wds_SLQSSetPacketSrvStatusCallback_t . . . 591
unpack_wds_SLQSSetWdsEventCallback_ind_t . . . 592
unpack_wds_SLQSSGetDHCPv4ClientConfig_t . . . 594
unpack_wds_SLQSSGetLoopback_t . . . 595
unpack_wds_SLQSStartDataSession_t . . . 595
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t . . . 596
UnPackGetProfileSettingOut . . . 597
unpackWdsProfileParam . . . 598
voice_airTimer . . . 598
voice_allCallsAlphaIDInfo . . . 598
voice_allCallsDiagInfo . . . 599
voice_allCallsUUSInfo . . . 599
voice_alphaIDInfo . . . 600
voice_arrAlertingPattern . . . 601
voice_arrAlertingType . . . 601
voice_arrAlphaID . . . 602
voice_arrCalledPartyNum . . . 603
voice_arrCallEndReason . . . 603
voice_arrCallInfo . . . 604
voice_arrConnectPartyNum . . . 604
voice_arrDiagInfo . . . 605
voice_arrRedirPartyNum . . . 605
voice_arrRemotePartyName . . . 606
voice_arrRemotePartyNum . . . 607
voice_arrSvcOption . . . 607
voice_arrUUSInfo . . . 608
voice_burstDTMFInfo . . . 608
voice_calledPartySubAdd . . . 609
voice_callFwdTypeAndPlan . . . 610
voice_callFWExtInfo . . . 611
voice_callFWInfo . . . 613
voice_callInfo . . . 614
voice_ccSUPSType . . . 615
voice_CLIPResp . . . 616
voice_CLIRResp . . . 617
voice_CNAPResp . . . 617
voice_COLPResp . . . 618
voice_COLRResp . . . 619
voice_connectNumInfo . . . 620
voice_CUGInfo . . . 621
voice_curAMRConfig . . . 622
voice_diagInfo . . . 623
voice_DTMFLengths . . . 623
voice_getAllCallInformation . . . 624
voice_getAllCallRmtPtyName . . . 625
voice_getAllCallRmtPtyNum . . . 625
voice_getCallFWExtInfo . . . 626
voice_getCallFWInfo . . . 626

voice_peerNumberInfo . . . 627
voice_prefVoiceSO . . . 628
voice_remotePartyName . . . 630
voice_remotePartyNum . . . 631
voice_roamTimer . . . 632
voice_USSInfo . . . 633
voice_UUSInfo . . . 633
wds_currNetworkInfo . . . 635
wds_DataULongLongTlv . . . 636
wds_DataULongTlv . . . 636
wds_DHCPLeaseOptTlv . . . 637
wds_DHCPLeaseStateTlv . . . 637
wds_DHCPOpt . . . 637
wds_DHCPProfileIdTlv . . . 638
wds_DHCPv4HWConfig . . . 638
wds_DHCPv4Option . . . 639
wds_DHCPv4OptionList . . . 639
wds_DHCPv4ProfileId . . . 640
wds_Domain . . . 640
wds_DomainNameList . . . 641
wds_GPRSQoS . . . 641
wds_IPv4AdTlv . . . 642
wds_IPV6AddressInfo . . . 642
wds_IPV6GWAddressInfo . . . 643
wds_PCSCFFQDNAddress . . . 643
wds_PCSCFFQDNAddressList . . . 644
wds_PCSCFIPv4ServerAddressList . . . 644
wds_ProfileIdentifier . . . 645
wds_profileInfo . . . 645
wds_TrStatInd . . . 646
wds_UMTSMinQoS . . . 646
wdsDhcpv4HwConfig . . . 649
wdsDhcpv4Option . . . 649
wdsDhcpv4OptionList . . . 650
wdsDhcpv4ProfileId . . . 650

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 . . . 653
common.h . . . 653
dms.h . . . 658
fms.h . . . 709
lite-fw.h . . . 712
loc.h . . . 719
nas.h . . . 734
pds.h . . . 776
qaGobiApiTableBandClasses.h
Network Access Service API Band Classes table . . . 781
qaGobiApiTableCallControlReturnReasons.h
Call Control Return Reasons table . . . 784
qaGobiApiTableCallEndReasons.h
Wireless Data Service Call End Reasons . . . 785
qaGobiApiTableCarrierCodes.h
Carrier Codes table . . . 800
qaGobiApiTableCodingScheme.h
Data Coding Scheme . . . 802
qaGobiApiTableGpsCapabilityCodes.h
Position Determination Service API GPS Capability Codes . . . 805
qaGobiApiTablePowerModes.h
Device Management Service API Power Modes table . . . 805
qaGobiApiTableRadioInterfaces.h
Network Access Service API Radio Interfaces table . . . 806
qaGobiApiTableRegionCodes.h
Region Codes table . . . 807
qaGobiApiTableServiceOptions.h
Voice Service Options . . . 807
qaGobiApiTableSupServiceInfoClasses.h
Voice Supplementary Service Information Classes . . . 809
qaGobiApiTableSwiAudio.h
Swi Audio related tables . . . 810
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h
Update Complete Status table . . . 810
qaGobiApiTableVoiceCallEndReasons.h
Voice Service Call and supplementary services end reasons . . . 812

qmerrno.h 818
qos.h 825
rms.h 833
sar.h 835
sms.h 837
swiavms.h 859
SwiDataTypes.h
SWI data types 877
swiloc.h 878
swioma.h 880
SWIWWANCMAPI.h 889
uim.h 889
voice.h 907
wds.h 934

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

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

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
UnpackSwiAvmsEventReportSessionType[OUT]	• See pSessionType for more information
pWAMSParaChanged[OUT]	• See UnpackSwiAvmsEventReportHTTPStatus for more information
pHTTPStatus[OUT]	• See pSessionType 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 cdmaSSInfo Struct Reference

Data Fields

- int8_t rssi
- int16_t ecio

8.23.1 Detailed Description

Parameters

rssi	RSSI in dBm.
ecio	ECIO value representing negative 0.5 dBm increment

8.23.2 Field Documentation

8.23.2.1 int16_t cdmaSSInfo::ecio

8.23.2.2 int8_t cdmaSSInfo::rssi

8.24 connectionStatus Struct Reference

Data Fields

- uint8_t MDMConnStatus
- uint64_t MDMCallDuration

8.24.1 Detailed Description

Parameters

MDMConn-Status	Current link status 1-Disconected 2-Connected
MDMCall-Duration	Call duration in milliseconds

8.24.2 Field Documentation

8.24.2.1 uint64_t connectionStatus::MDMCallDuration

8.24.2.2 uint8_t connectionStatus::MDMConnStatus

8.25 crashInfoParams Struct Reference

Data Fields

- uint8_t crashStatus
- crashInformation crashInfo

8.25.1 Detailed Description

This structure contains crashInfoParams

Parameters

Tlvresult	• Unpack Result
crashInfoParam[-OUT]	• See crashInfoParams

8.25.2 Field Documentation

8.25.2.1 crashInformation crashInfoParams::crashInfo

8.25.2.2 uint8_t crashInfoParams::crashStatus

8.26 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]

8.26.1 Detailed Description

This structure contains crashInformation

Parameters

numCrashes[O-UT]	• 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[OUT]	• length of the gcdumpString field as an input, length of the gcdumpString field returned by API as an OUTPUT
gcdumpString[OUT]	• gcdump string for the crash

8.26.2 Field Documentation

8.26.2.1 uint32_t crashInformation::crashData

8.26.2.2 uint32_t crashInformation::crashId

8.26.2.3 char crashInformation::crashString[255]

8.26.2.4 uint16_t crashInformation::crashStrlen

8.26.2.5 char crashInformation::gcdumpString[1024]

8.26.2.6 uint16_t crashInformation::gcdumpStrlen

8.26.2.7 uint16_t crashInformation::numCrashes

8.27 currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.27.1 Detailed Description

Parameters

NetworkType	network type 0-3GPP 1-3GPP2
RATMask	Radio access technology mask to indicate technology.
SOMask	Service Option mask to indicate the service option.

8.27.2 Field Documentation

8.27.2.1 uint8_t currNetworkInfo::NetworkType

8.27.2.2 uint32_t currNetworkInfo::RATMask

8.27.2.3 uint32_t currNetworkInfo::SOMask

8.28 dms_ActivationStatusTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t activationStatus

8.28.1 Detailed Description

Activation Status Tlv

Parameters

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

8.28.2 Field Documentation

8.28.2.1 uint32_t dms_ActivationStatusTlv::activationStatus

8.28.2.2 uint16_t dms_ActivationStatusTlv::TlvPresent

8.29 dms_OperatingModeTlv Struct Reference

Data Fields

- uint16_t TlvPresent
- uint32_t operatingMode

8.29.1 Detailed Description

Operating Mode Tlv

Parameters

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

8.29.2 Field Documentation

8.29.2.1 uint32_t dms_OperatingModeTlv::operatingMode

8.29.2.2 uint16_t dms_OperatingModeTlv::TlvPresent

8.30 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

8.30.1 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)
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

8.30.2 Field Documentation

8.30.2.1 uint16_t DMScustSettingInfo::cust_attr

8.30.2.2 uint8_t DMScustSettingInfo::cust_id[64+1]

8.30.2.3 uint8_t DMScustSettingInfo::cust_value[8+1]

8.30.2.4 uint16_t DMScustSettingInfo::id_length

8.30.2.5 uint16_t DMScustSettingInfo::value_length

8.31 DMScustSettingList Struct Reference

Data Fields

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

8.31.1 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 custSettingInfo for more information

8.31.2 Field Documentation

8.31.2.1 DMScustSettingInfo DMScustSettingList::custSetting[255+1]

8.31.2.2 uint8_t DMScustSettingList::list_type

8.31.2.3 uint16_t DMScustSettingList::num_instances

8.32 DMSgetCustomFeatureV2 Struct Reference

Data Fields

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

8.32.1 Detailed Description

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

Parameters

IN]	pGetCustomInput • See getCustomInput for more information
OUT]	pCustSettingInfo • See custSettingInfo for more information
OUT]	pCustSettingList • See custSettingList for more information

8.32.2 Field Documentation

8.32.2.1 DMScustSettingInfo* DMSgetCustomFeatureV2::pCustSettingInfo

8.32.2.2 DMScustSettingList* DMSgetCustomFeatureV2::pCustSettingList

8.32.2.3 DMSgetCustomInput* DMSgetCustomFeatureV2::pGetCustomInput

8.33 DMSgetCustomInput Struct Reference

Data Fields

- uint8_t cust_id [64+1]
- uint8_t list_type

8.33.1 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)
list_type	• list type requested

8.33.2 Field Documentation

8.33.2.1 uint8_t DMSgetCustomInput::cust_id[64+1]

8.33.2.2 uint8_t DMSgetCustomInput::list_type

8.34 dunchannelRate Struct Reference

Data Fields

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

8.34.1 Detailed Description

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
MaxChanRx-Rate	maximum Rx rate that can be assigned to the device

8.34.2 Field Documentation

8.34.2.1 uint32_t dunchannelRate::CurrChanRxRate

8.34.2.2 uint32_t dunchannelRate::CurrChanTxRate

8.34.2.3 uint32_t dunchannelRate::MaxChanRxRate

8.34.2.4 uint32_t dunchannelRate::MaxChanTxRate

8.35 eriDataparams Struct Reference

Data Fields

- uint16_t eriDataLen
- uint8_t eriData [1024]

8.35.1 Field Documentation

8.35.1.1 uint8_t eriDataparams::eriData[1024]

8.35.1.2 uint16_t eriDataparams::eriDataLen

8.36 eTWSPLMNInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsPlmnInfo ETWSPLMNInfo

8.36.1 Detailed Description

Parameters

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

8.36.2 Field Documentation

8.36.2.1 sMSEtwsPlmnInfo eTWSPLMNInfoTlv::ETWSPLMNInfo

8.36.2.2 uint8_t eTWSPLMNInfoTlv::TlvPresent

8.37 FMSImageElement Struct Reference

Data Fields

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

8.37.1 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

8.37.2 Field Documentation

8.37.2.1 uint8_t FMSImageElement::buildId[100]

8.37.2.2 uint8_t FMSImageElement::buildIdLength

8.37.2.3 uint8_t FMSImageElement::imageId[16]

8.37.2.4 uint8_t FMSImageElement::imageType

8.38 FMSImageIdElement Struct Reference

Data Fields

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

8.38.1 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)

8.38.2 Field Documentation

8.38.2.1 uint8_t FMSImageIdElement::buildID[100]

8.38.2.2 uint8_t FMSImageIdElement::buildIDLength

8.38.2.3 uint8_t FMSImageIdElement::failureCount

8.38.2.4 uint8_t FMSImageIdElement::imageID[16]

8.38.2.5 uint8_t FMSImageIdElement::storageIndex

8.39 FMSImageIDEntries Struct Reference

Data Fields

- uint8_t imageType
- uint8_t maxImages
- uint8_t executingImage
- uint8_t imageIDSize

- FMSImageIdElement imageIDElement [50]

8.39.1 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

8.39.2 Field Documentation

8.39.2.1 uint8_t FMSImageIDEntries::executingImage

8.39.2.2 FMSImageIdElement FMSImageIDEntries::imageIDElement[50]

8.39.2.3 uint8_t FMSImageIDEntries::imageIDSize

8.39.2.4 uint8_t FMSImageIDEntries::imageType

8.39.2.5 uint8_t FMSImageIDEntries::maxImages

8.40 FMSImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageIDEntries imageIDEntries [2]

8.40.1 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)

8.40.2 Field Documentation

8.40.2.1 FMSImageIDEntries FMSImageList::imageIDEntries[2]

8.40.2.2 uint8_t FMSImageList::listSize

8.41 FMSPrefImageList Struct Reference

Data Fields

- uint8_t listSize
- FMSImageElement listEntries [2]

8.41.1 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

8.41.2 Field Documentation

8.41.2.1 FMSImageElement FMSPrefImageList::listEntries[2]

8.41.2.2 uint8_t FMSPrefImageList::listSize

8.42 hdrSSInfo Struct Reference

Data Fields

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

8.42.1 Detailed Description

Parameters

rssi	RSSI in dBm.
ecio	ECIO value representing negative 0.5 dBm increment
sinr	SINR level.
io	Received IO in dBm.

8.42.2 Field Documentation

8.42.2.1 int16_t hdrSSInfo::ecio

8.42.2.2 int32_t hdrSSInfo::io

8.42.2.3 int8_t hdrSSInfo::rssi

8.42.2.4 uint8_t hdrSSInfo::sinr

8.43 image_info_t Struct Reference

Data Fields

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

8.43.1 Field Documentation

8.43.1.1 uint8_t image_info_t::buildID[255]

8.43.1.2 uint8_t image_info_t::buildIDLen

8.43.1.3 uint8_t image_info_t::imageType

8.43.1.4 uint8_t image_info_t::uniqueID[16]

8.44 ipv6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.44.1 Detailed Description

Parameters

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

8.44.2 Field Documentation

8.44.2.1 uint16_t ipv6AddressInfo::IPAddressV6[8]

8.44.2.2 uint8_t ipv6AddressInfo::IPV6PrefixLen

8.45 LibPackGPRSRequestedQoS Struct Reference

Data Fields

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

8.45.1 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

8.45.2 Field Documentation

8.45.2.1 uint32_t LibPackGPRSRequestedQoS::delayClass

8.45.2.2 uint32_t LibPackGPRSRequestedQoS::meanThroughputClass

8.45.2.3 uint32_t LibPackGPRSRequestedQoS::peakThroughputClass

8.45.2.4 uint32_t LibPackGPRSRequestedQoS::precedenceClass

8.45.2.5 uint32_t LibPackGPRSRequestedQoS::reliabilityClass

8.46 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

8.46.1 Detailed Description

Parameters

extended	error
profile	info This structure contains Input parameters of SLQSCreateProfile and SLQSModifyProfile and output parameters of SLQSGetProfileSettings • Parameter values default to their data type's maximum unsigned value unless explicitly stated otherwise.
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
pPdpHdrCompType	• 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 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().

8.46.2 Field Documentation

8.46.2.1 uint8_t* LibpackProfile3GPP::pAddrAllocPref

8.46.2.2 uint8_t* LibpackProfile3GPP::pAPNClass

8.46.2.3 uint8_t* LibpackProfile3GPP::pAPNDisabledFlag

8.46.2.4 uint8_t* LibpackProfile3GPP::pAPNName

8.46.2.5 uint16_t* LibpackProfile3GPP::pAPNnameSize

8.46.2.6 uint8_t* LibpackProfile3GPP::pAuthenticationPref

8.46.2.7 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSMinimumQoS

8.46.2.8 LibPackGPRSRequestedQoS* LibpackProfile3GPP::pGPRSRequestedQos

8.46.2.9 uint8_t* LibpackProfile3GPP::pImCnFlag

8.46.2.10 uint32_t* LibpackProfile3GPP::pIPv4AddrPref

8.46.2.11 uint16_t* LibpackProfile3GPP::pIPv6AddPref

8.46.2.12 uint8_t* LibpackProfile3GPP::pPassword

8.46.2.13 uint16_t* LibpackProfile3GPP::pPasswordSize

8.46.2.14 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingDhcp

8.46.2.15 uint8_t* LibpackProfile3GPP::pPcscfAddrUsingPCO

8.46.2.16 uint32_t* LibpackProfile3GPP::pPDNInactivTimeout

8.46.2.17 uint8_t* LibpackProfile3GPP::pPdpAccessConFlag

8.46.2.18 uint8_t* LibpackProfile3GPP::pPdpContext

8.46.2.19 uint8_t* LibpackProfile3GPP::pPdpDataCompType

8.46.2.20 uint8_t* LibpackProfile3GPP::pPdpHdrCompType

8.46.2.21 uint8_t* LibpackProfile3GPP::pPDPtype

8.46.2.22 uint32_t* LibpackProfile3GPP::pPriDNSIPv4AddPref

8.46.2.23 uint16_t* LibpackProfile3GPP::pPriDNSIPv6addpref

8.46.2.24 uint8_t* LibpackProfile3GPP::pPrimaryID

8.46.2.25 uint8_t* LibpackProfile3GPP::pProfilename

8.46.2.26 uint16_t* LibpackProfile3GPP::pProfilenameSize

8.46.2.27 LibPackQosClassID* LibpackProfile3GPP::pQosClassID

8.46.2.28 uint32_t* LibpackProfile3GPP::pSecDNSIPv4AddPref

8.46.2.29 uint16_t* LibpackProfile3GPP::pSecDNSIPv6addpref

8.46.2.30 uint8_t* LibpackProfile3GPP::pSecondaryFlag

8.46.2.31 LibPackTFTIDParams* LibpackProfile3GPP::pTFTID1Params

8.46.2.32 LibPackTFTIDParams* LibpackProfile3GPP::pTFTID2Params

8.46.2.33 LibPackUMTSQoS* LibpackProfile3GPP::pUMTSMinQoS

8.46.2.34 LibPackUMTSReqQoSSigInd* LibpackProfile3GPP::pUMTSMinQosSigInd

8.46.2.35 LibPackUMTSQoS* LibpackProfile3GPP::pUMTSReqQoS

8.46.2.36 LibPackUMTSReqQoSSigInd* LibpackProfile3GPP::pUMTSReqQoSSigInd

8.46.2.37 uint8_t* LibpackProfile3GPP::pUsername

8.46.2.38 uint16_t* LibpackProfile3GPP::pUsernameSize

8.47 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

8.47.1 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 SLQSGetProfile-Settings().
pAPNEnabled3-GPP2	• 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 SLQSGetProfile-Settings().
pPDNInactiv-Timeout3GPP2	• 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().
pAPNClass3GP-P2	• 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().

8.47.2 Field Documentation

8.47.2.1 uint8_t* LibpackProfile3GPP2::pAllowLinger

8.47.2.2 uint8_t* LibpackProfile3GPP2::pAPNClass3GPP2

8.47.2.3 uint8_t* LibpackProfile3GPP2::pAPNEnabled3GPP2

8.47.2.4 uint8_t* LibpackProfile3GPP2::pApnString

8.47.2.5 uint16_t* LibpackProfile3GPP2::pApnStringSize

8.47.2.6 uint8_t* LibpackProfile3GPP2::pAppPriority

8.47.2.7 uint32_t* LibpackProfile3GPP2::pAppType

8.47.2.8 uint8_t* LibpackProfile3GPP2::pAuthPassword

8.47.2.9 uint16_t* LibpackProfile3GPP2::pAuthPasswordSize

8.47.2.10 uint8_t* LibpackProfile3GPP2::pAuthProtocol

8.47.2.11 uint8_t* LibpackProfile3GPP2::pAuthRetryCount

8.47.2.12 uint16_t* LibpackProfile3GPP2::pAuthTimeout

8.47.2.13 uint8_t* LibpackProfile3GPP2::pDataMode

8.47.2.14 uint8_t* LibpackProfile3GPP2::pDataRate

8.47.2.15 uint16_t* LibpackProfile3GPP2::pIpcpAckTimeout

8.47.2.16 uint8_t* LibpackProfile3GPP2::pIpcpCreqRetryCount

8.47.2.17 uint8_t* LibpackProfile3GPP2::pIsPcscfAddressNedded

8.47.2.18 uint16_t* LibpackProfile3GPP2::pLcpAckTimeout

8.47.2.19 uint8_t* LibpackProfile3GPP2::pLcpCreqRetryCount

8.47.2.20 uint8_t* LibpackProfile3GPP2::pNegoDnsSrvrPref

8.47.2.21 uint32_t* LibpackProfile3GPP2::pPDNInactivTimeout3GPP2

8.47.2.22 uint8_t* LibpackProfile3GPP2::pPdnType

8.47.2.23 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimer1x

8.47.2.24 uint32_t* LibpackProfile3GPP2::pPppSessCloseTimerDO

8.47.2.25 uint32_t* LibpackProfile3GPP2::pPrimaryV4DnsAddress

8.47.2.26 uint16_t* LibpackProfile3GPP2::pPriV6DnsAddress

8.47.2.27 uint8_t* LibpackProfile3GPP2::pRATType

8.47.2.28 uint32_t* LibpackProfile3GPP2::pSecondaryV4DnsAddress

8.47.2.29 uint16_t* LibpackProfile3GPP2::pSecV6DnsAddress

8.47.2.30 uint8_t* LibpackProfile3GPP2::pUserId

8.47.2.31 uint16_t* LibpackProfile3GPP2::pUserIdSize

8.48 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

8.48.1 Detailed Description

This structure contains Input parameters of SLQSCreateProfile and SLQSModifyProfile and output parameters of SLQSGetProfileSettings

- 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().

8.48.2 Field Documentation

8.48.2.1 uint8_t* LibPackprofile_3GPP::pAddrAllocPref

8.48.2.2 uint8_t* LibPackprofile_3GPP::pAPNClass

8.48.2.3 uint8_t* LibPackprofile_3GPP::pAPNDisabledFlag

8.48.2.4 uint8_t* LibPackprofile_3GPP::pAPNName

8.48.2.5 uint16_t* LibPackprofile_3GPP::pAPNnameSize

8.48.2.6 uint8_t* LibPackprofile_3GPP::pAuthenticationPref

8.48.2.7 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSMinimumQoS

8.48.2.8 LibPackGPRSRequestedQoS* LibPackprofile_3GPP::pGPRSRequestedQos

8.48.2.9 uint8_t* LibPackprofile_3GPP::pImCnFlag

8.48.2.10 uint32_t* LibPackprofile_3GPP::pIPv4AddrPref

8.48.2.11 uint16_t* LibPackprofile_3GPP::pIPv6AddPref

8.48.2.12 uint8_t* LibPackprofile_3GPP::pPassword

8.48.2.13 uint16_t* LibPackprofile_3GPP::pPasswordSize

8.48.2.14 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingDhcp

8.48.2.15 uint8_t* LibPackprofile_3GPP::pPcscfAddrUsingPCO

8.48.2.16 uint32_t* LibPackprofile_3GPP::pPDNInactivTimeout

8.48.2.17 uint8_t* LibPackprofile_3GPP::pPdpAccessConFlag

8.48.2.18 uint8_t* LibPackprofile_3GPP::pPdpContext

8.48.2.19 uint8_t* LibPackprofile_3GPP::pPdpDataCompType

8.48.2.20 uint8_t* LibPackprofile_3GPP::pPdpHdrCompType

8.48.2.21 uint8_t* LibPackprofile_3GPP::pPDPtype

8.48.2.22 uint32_t* LibPackprofile_3GPP::pPriDNSIPv4AddPref

8.48.2.23 uint16_t* LibPackprofile_3GPP::pPriDNSIPv6addpref

8.48.2.24 uint8_t* LibPackprofile_3GPP::pPrimaryID

8.48.2.25 uint8_t* LibPackprofile_3GPP::pProfilename

8.48.2.26 uint16_t* LibPackprofile_3GPP::pProfilenameSize

8.48.2.27 **LibPackQosClassID*** LibPackprofile_3GPP::pQosClassID

8.48.2.28 uint32_t* LibPackprofile_3GPP::pSecDNSIPv4AddPref

8.48.2.29 uint16_t* LibPackprofile_3GPP::pSecDNSIPv6addpref

8.48.2.30 uint8_t* LibPackprofile_3GPP::pSecondaryFlag

8.48.2.31 **LibPackTFTIDParams*** LibPackprofile_3GPP::pTFTID1Params

8.48.2.32 **LibPackTFTIDParams*** LibPackprofile_3GPP::pTFTID2Params

8.48.2.33 **LibPackUMTSQoS*** LibPackprofile_3GPP::pUMTSMinQoS

8.48.2.34 **LibPackUMTSReqQoSSigInd*** LibPackprofile_3GPP::pUMTSMinQosSigInd

8.48.2.35 **LibPackUMTSQoS*** LibPackprofile_3GPP::pUMTSReqQoS

8.48.2.36 **LibPackUMTSReqQoSSigInd*** LibPackprofile_3GPP::pUMTSReqQoSSigInd

8.48.2.37 uint8_t* LibPackprofile_3GPP::pUsername

8.48.2.38 uint16_t* LibPackprofile_3GPP::pUsernameSize

8.49 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

8.49.1 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 SLQSGetProfile-Settings().
pAPNEnabled3-GPP2	• 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 SLQSGetProfile-Settings().
pPDNInactiv-Timeout3GPP2	• 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().
pAPNClass3GP-P2	• 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().

8.49.2 Field Documentation

8.49.2.1 uint8_t* LibPackprofile_3GPP2::pAllowLinger

8.49.2.2 uint8_t* LibPackprofile_3GPP2::pAPNClass3GPP2

8.49.2.3 uint8_t* LibPackprofile_3GPP2::pAPNEnabled3GPP2

8.49.2.4 uint8_t* LibPackprofile_3GPP2::pApnString

8.49.2.5 uint16_t* LibPackprofile_3GPP2::pApnStringSize

8.49.2.6 uint8_t* LibPackprofile_3GPP2::pAppPriority

8.49.2.7 uint32_t* LibPackprofile_3GPP2::pAppType

8.49.2.8 uint8_t* LibPackprofile_3GPP2::pAuthPassword

8.49.2.9 uint16_t* LibPackprofile_3GPP2::pAuthPassword_tSize

8.49.2.10 uint8_t* LibPackprofile_3GPP2::pAuthProtocol

8.49.2.11 uint8_t* LibPackprofile_3GPP2::pAuthRetryCount

8.49.2.12 uint16_t* LibPackprofile_3GPP2::pAuthTimeout

8.49.2.13 uint8_t* LibPackprofile_3GPP2::pDataMode

8.49.2.14 uint8_t* LibPackprofile_3GPP2::pDataRate

8.49.2.15 uint16_t* LibPackprofile_3GPP2::pIpcpAckTimeout

8.49.2.16 uint8_t* LibPackprofile_3GPP2::pIpcpCreqRetryCount

8.49.2.17 uint8_t* LibPackprofile_3GPP2::pIsPcscfAddressNedded

8.49.2.18 uint16_t* LibPackprofile_3GPP2::pLcpAckTimeout

8.49.2.19 uint8_t* LibPackprofile_3GPP2::pLcpCreqRetryCount

8.49.2.20 uint8_t* LibPackprofile_3GPP2::pNegoDnsSrvrPref

8.49.2.21 uint32_t* LibPackprofile_3GPP2::pPDNInactivTimeout3GPP2

8.49.2.22 uint8_t* LibPackprofile_3GPP2::pPdnType

8.49.2.23 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimer1x

8.49.2.24 uint32_t* LibPackprofile_3GPP2::pPppSessCloseTimerDO

8.49.2.25 uint32_t* LibPackprofile_3GPP2::pPrimaryV4DnsAddress

8.49.2.26 uint16_t* LibPackprofile_3GPP2::pPriV6DnsAddress

8.49.2.27 uint8_t* LibPackprofile_3GPP2::pRATType

8.49.2.28 uint32_t* LibPackprofile_3GPP2::pSecondaryV4DnsAddress

8.49.2.29 uint16_t* LibPackprofile_3GPP2::pSecV6DnsAddress

8.49.2.30 uint8_t* LibPackprofile_3GPP2::pUserId

8.49.2.31 uint16_t* LibPackprofile_3GPP2::pUserIdSize

8.50 LibPackQosClassID Struct Reference

Data Fields

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

8.50.1 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	• maxDlBitRate
gUlBitRate	• Guaranteed UL bit rate
maxUlBitRate	• Maximum UL bit rate

8.50.2 Field Documentation

8.50.2.1 uint8_t LibPackQosClassID::gDlBitRate

8.50.2.2 uint32_t LibPackQosClassID::gUlBitRate

8.50.2.3 uint32_t LibPackQosClassID::maxDlBitRate

8.50.2.4 uint32_t LibPackQosClassID::maxUlBitRate

8.50.2.5 uint8_t LibPackQosClassID::QCI

8.51 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

8.51.1 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
pVersion	• IP version number – 4 - IPv4 – 6 - IPv6
sourceIP	• 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

8.51.2 Field Documentation

8.51.2.1 uint16_t LibPackTFTIDParams::destPortRangeEnd

8.51.2.2 uint32_t LibPackTFTIDParams::destPortRangeStart

8.51.2.3 uint8_t LibPackTFTIDParams::eValid

8.51.2.4 uint8_t LibPackTFTIDParams::filterId

8.51.2.5 uint32_t LibPackTFTIDParams::flowLabel

8.51.2.6 uint32_t LibPackTFTIDParams::IPSECSPI

8.51.2.7 uint8_t LibPackTFTIDParams::ipVersion

8.51.2.8 uint8_t LibPackTFTIDParams::nextHeader

8.51.2.9 uint16_t* LibPackTFTIDParams::pSourceIP

8.51.2.10 uint8_t LibPackTFTIDParams::sourceIPMask

8.51.2.11 uint16_t LibPackTFTIDParams::srcPortRangeEnd

8.51.2.12 uint16_t LibPackTFTIDParams::srcPortRangeStart

8.51.2.13 uint16_t LibPackTFTIDParams::tosMask

8.52 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

8.52.1 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.

8.52.2 Field Documentation

8.52.2.1 uint8_t LibPackUMTSQoS::deliveryErrSDU

8.52.2.2 uint32_t LibPackUMTSQoS::grntDownlinkBitrate

8.52.2.3 uint32_t LibPackUMTSQoS::grntUplinkBitrate

8.52.2.4 uint32_t LibPackUMTSQoS::maxDownlinkBitrate

8.52.2.5 uint32_t LibPackUMTSQoS::maxSDUSize

8.52.2.6 uint32_t LibPackUMTSQoS::maxUplinkBitrate

8.52.2.7 uint8_t LibPackUMTSQoS::qosDeliveryOrder

8.52.2.8 uint8_t LibPackUMTSQoS::resBerRatio

8.52.2.9 uint8_t LibPackUMTSQoS::sduErrorRatio

8.52.2.10 uint8_t LibPackUMTSQoS::trafficClass

8.52.2.11 uint32_t LibPackUMTSQoS::trafficPriority

8.52.2.12 uint32_t LibPackUMTSQoS::transferDelay

8.53 LibPackUMTSReqQoSSigInd Struct Reference

Data Fields

- LibPackUMTSQoS UMTSReqQoS
- uint8_t SigInd

8.53.1 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

8.53.2 Field Documentation

8.53.2.1 uint8_t LibPackUMTSReqQoSSigInd::SigInd

8.53.2.2 LibPackUMTSQoS LibPackUMTSReqQoSSigInd::UMTSReqQoS

8.54 loc_BdsSV Struct Reference

Data Fields

- uint16_t id
- uint8_t mask

8.54.1 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

8.54.2 Field Documentation

8.54.2.1 uint16_t loc_BdsSV::id

8.54.2.2 uint8_t loc_BdsSV::mask

8.55 loc_BdsSVInfo Struct Reference

Data Fields

- uint8_t len
- loc_BdsSV * pSV

8.55.1 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

8.55.2 Field Documentation

8.55.2.1 uint8_t loc_BdsSVInfo::len

8.55.2.2 loc_BdsSV* loc_BdsSVInfo::pSV

8.56 loc_CellDb Struct Reference

Data Fields

- uint32_t mask

8.56.1 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

8.56.2 Field Documentation

8.56.2.1 uint32_t loc_CellDb::mask

8.57 loc_ClkInfo Struct Reference

Data Fields

- uint32_t mask

8.57.1 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

8.57.2 Field Documentation

8.57.2.1 uint32_t loc_ClkInfo::mask

8.58 loc_GnssData Struct Reference

Data Fields

- uint64_t mask

8.58.1 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

8.58.2 Field Documentation

8.58.2.1 uint64_t loc_GnssData::mask

8.59 loc_gpsTime Struct Reference

Data Fields

- uint16_t gpsWeek
- uint32_t gpsTimeOfWeekMs

8.59.1 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

8.59.2 Field Documentation

8.59.2.1 uint32_t loc_gpsTime::gpsTimeOfWeekMs

8.59.2.2 uint16_t loc_gpsTime::gpsWeek

8.60 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

8.60.1 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

8.60.2 Field Documentation

8.60.2.1 uint8_t loc_LocApplicationInfo::appNameLength

8.60.2.2 uint8_t loc_LocApplicationInfo::appProviderLength

8.60.2.3 uint8_t loc_LocApplicationInfo::appVersionLength

8.60.2.4 uint8_t loc_LocApplicationInfo::appVersionValid

8.60.2.5 uint8_t* loc_LocApplicationInfo::pAppName

8.60.2.6 uint8_t* loc_LocApplicationInfo::pAppProvider

8.60.2.7 uint8_t* loc_LocApplicationInfo::pAppVersion

8.61 loc_precisionDilution Struct Reference

Data Fields

- uint32_t PDOP
- uint32_t HDOP
- uint32_t VDOP

8.61.1 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)

8.61.2 Field Documentation

8.61.2.1 uint32_t loc_precisionDilution::HDOP

8.61.2.2 uint32_t loc_precisionDilution::PDOP

8.61.2.3 uint32_t loc_precisionDilution::VDOP

8.62 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

8.62.1 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

8.62.2 Field Documentation

8.62.2.1 float loc_satelliteInfo::azimuth

8.62.2.2 float loc_satelliteInfo::elevation

8.62.2.3 uint16_t loc_satelliteInfo::gnssSvId

8.62.2.4 uint8_t loc_satelliteInfo::healthStatus

8.62.2.5 float loc_satelliteInfo::snr

8.62.2.6 uint8_t loc_satelliteInfo::svInfoMask

8.62.2.7 uint8_t loc_satelliteInfo::svListLen

8.62.2.8 uint32_t loc_satelliteInfo::svStatus

8.62.2.9 uint32_t loc_satelliteInfo::system

8.62.2.10 uint32_t loc_satelliteInfo::validMask

8.63 loc_sensorDataUsage Struct Reference

Data Fields

- uint32_t usageMask
- uint32_t aidingIndicatorMask

8.63.1 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

Parameters

aidingIndicator-Mask	

  - Specifies which results were aided by sensors.

- Value
  - 0x00000001 - AIDED_HEADING
  - 0x00000002 - AIDED_SPEED
  - 0x00000004 - AIDED_POSITION
  - 0x00000008 - AIDED_VELOCITY

8.63.2 Field Documentation

8.63.2.1 uint32_t loc_sensorDataUsage::aidingIndicatorMask

8.63.2.2 uint32_t loc_sensorDataUsage::usageMask

8.64 loc_SV Struct Reference

Data Fields

- uint16_t id
- uint32_t system
- uint8_t mask

8.64.1 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

8.64.2 Field Documentation

8.64.2.1 uint16_t loc_SV::id

8.64.2.2 uint8_t loc_SV::mask

8.64.2.3 uint32_t loc_SV::system

8.65 loc_SVInfo Struct Reference

Data Fields

- uint8_t len
- loc_SV * pSV

8.65.1 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

8.65.2 Field Documentation

8.65.2.1 uint8_t loc_SVInfo::len

8.65.2.2 loc_SV* loc_SVInfo::pSV

8.66 loc_svUsedforFix Struct Reference

Data Fields

- uint8_t gnssSvUsedList_len
- uint16_t gnssSvUsedList [255]

8.66.1 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

8.66.2 Field Documentation

8.66.2.1 uint16_t loc_svUsedforFix::gnssSvUsedList[255]

8.66.2.2 uint8_t loc_svUsedforFix::gnssSvUsedList_len

8.67 lteSSInfo Struct Reference

Data Fields

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

8.67.1 Detailed Description

Parameters

rssi	RSSI in dBm.
rsrq	RSRQ value in dB
rsrp	Current RSRP in dBm as measured by L1.
snr	SNR level as a scaled integer in units of 0.1 dB.

8.67.2 Field Documentation

8.67.2.1 int16_t lteSSInfo::rsrp

8.67.2.2 int8_t lteSSInfo::rsrq

8.67.2.3 int8_t lteSSInfo::rssi

8.67.2.4 int16_t lteSSInfo::snr

8.68 messageModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMessageModeInfo MessageModeInfo

8.68.1 Detailed Description

Parameters

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

8.68.2 Field Documentation

8.68.2.1 sMSMessageModeInfo messageModeTlv::MessageModeInfo

8.68.2.2 uint8_t messageModeTlv::TlvPresent

8.69 nas_acqOrderPref Struct Reference

Data Fields

- uint8_t acqOrdeLen
- uint8_t * pAcqOrder

8.69.1 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

8.69.2 Field Documentation

8.69.2.1 uint8_t nas_acqOrderPref::acqOrdeLen

8.69.2.2 uint8_t* nas_acqOrderPref::pAcqOrder

8.70 nas_ActPilotPNElement Struct Reference

Data Fields

- uint16_t ActSetPilotPN
- uint8_t ActSetPilotPNStrength

8.70.1 Detailed Description

Parameters

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

8.70.2 Field Documentation

8.70.2.1 uint16_t nas_ActPilotPNElement::ActSetPilotPN

8.70.2.2 uint8_t nas_ActPilotPNElement::ActSetPilotPNStrength

8.71 nas_AddCDMASysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx
- uint16_t regPrd

8.71.1 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.

8.71.2 Field Documentation

8.71.2.1 uint16_t nas_AddCDMASysInfo::geoSysIdx

8.71.2.2 uint16_t nas_AddCDMASysInfo::regPrd

8.72 nas_AddSysInfo Struct Reference

Data Fields

- uint16_t geoSysIdx
- uint32_t cellBroadcastCap

8.72.1 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

8.72.2 Field Documentation

8.72.2.1 uint32_t nas_AddSysInfo::cellBroadcastCap

8.72.2.2 uint16_t nas_AddSysInfo::geoSysIdx

8.73 nas_CallBarringSysInfo Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.73.1 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

8.73.2 Field Documentation

8.73.2.1 uint32_t nas_CallBarringSysInfo::csBarStatus

8.73.2.2 uint32_t nas_CallBarringSysInfo::psBarStatus

8.74 nas_callBarStatus Struct Reference

Data Fields

- uint32_t csBarStatus
- uint32_t psBarStatus

8.74.1 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

8.74.2 Field Documentation

8.74.2.1 uint32_t nas_callBarStatus::csBarStatus

8.74.2.2 uint32_t nas_callBarStatus::psBarStatus

8.75 nas_CDMAChannel Struct Reference

Data Fields

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

8.75.1 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

8.75.2 Field Documentation

8.75.2.1 uint16_t nas_CDMAChannel::priChA

8.75.2.2 uint16_t nas_CDMAChannel::priChB

8.75.2.3 uint16_t nas_CDMAChannel::secChA

8.75.2.4 uint16_t nas_CDMAChannel::secChB

8.76 nas_CDMAECIOThresh Struct Reference

Data Fields

- uint8_t CDMAECIOThreshListLen
- int16_t * pCDMAECIOThreshList

8.76.1 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)

8.76.2 Field Documentation

8.76.2.1 uint8_t nas_CDMAECIOThresh::CDMAECIOThreshListLen

8.76.2.2 int16_t* nas_CDMAECIOThresh::pCDMAECIOThreshList

8.77 nas_CDMAInfo Struct Reference

Data Fields

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

- uint32_t baseLong

8.77.1 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

8.77.2 Field Documentation

8.77.2.1 uint16_t nas_CDMAInfo::baseId

8.77.2.2 uint32_t nas_CDMAInfo::baseLat

8.77.2.3 uint32_t nas_CDMAInfo::baseLong

8.77.2.4 uint16_t nas_CDMAInfo::nid

8.77.2.5 uint16_t nas_CDMAInfo::refpn

8.77.2.6 uint16_t nas_CDMAInfo::sid

8.78 nas_CDMARSSIThresh Struct Reference

Data Fields

- uint8_t CDMARSSIThreshListLen
- int16_t * pCDMARSSIThreshList

8.78.1 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).

8.78.2 Field Documentation

8.78.2.1 uint8_t nas_CDMARSSIThresh::CDMARSSIThreshListLen

8.78.2.2 int16_t* nas_CDMARSSIThresh::pCDMARSSIThreshList

8.79 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]

8.79.1 Detailed Description

Structure for storing the CDMA System Information.

Parameters

sysInfoCDMA	• See 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}.

8.79.2 Field Documentation

8.79.2.1 uint16_t nas_CDMASysInfo::baseId

8.79.2.2 uint32_t nas_CDMASysInfo::baseLat

8.79.2.3 uint32_t nas_CDMASysInfo::baseLong

8.79.2.4 uint8_t nas_CDMASysInfo::bsInfoValid

8.79.2.5 uint8_t nas_CDMASysInfo::bsPRev

8.79.2.6 uint8_t nas_CDMASysInfo::bsPRevValid

8.79.2.7 uint8_t nas_CDMASysInfo::ccsSupported

8.79.2.8 uint8_t nas_CDMASysInfo::ccsSupportedValid

8.79.2.9 uint8_t nas_CDMASysInfo::cdmaSysIdValid

8.79.2.10 uint8_t nas_CDMASysInfo::isSysPrlMatch

8.79.2.11 uint8_t nas_CDMASysInfo::isSysPrlMatchValid

8.79.2.12 uint8_t nas_CDMASysInfo::MCC[3]

8.79.2.13 uint8_t nas_CDMASysInfo::MNC[3]

8.79.2.14 uint16_t nas_CDMASysInfo::networkID

8.79.2.15 uint8_t nas_CDMASysInfo::networkIdValid

8.79.2.16 uint16_t nas_CDMASysInfo::packetZone

8.79.2.17 uint8_t nas_CDMASysInfo::packetZoneValid

8.79.2.18 uint8_t nas_CDMASysInfo::pRevInUse

8.79.2.19 uint8_t nas_CDMASysInfo::pRevInUseValid

8.79.2.20 nas_sysInfoCommon nas_CDMASysInfo::sysInfoCDMA

8.79.2.21 uint16_t nas_CDMASysInfo::systemID

8.80 nas_CDMASysInfoExt Struct Reference

Data Fields

- uint16_t MCC
- uint8_t imsi_11_12

8.80.1 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

8.80.2 Field Documentation

8.80.2.1 uint8_t nas_CDMASysInfoExt::imsi_11_12

8.80.2.2 uint16_t nas_CDMASysInfoExt::MCC

8.81 nas_cellParams Struct Reference

Data Fields

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

8.81.1 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.

8.81.2 Field Documentation

8.81.2.1 uint16_t nas_cellParams::pci

8.81.2.2 int16_t nas_cellParams::rsrp

8.81.2.3 int16_t nas_cellParams::rsrq

8.81.2.4 int16_t nas_cellParams::rssi

8.81.2.5 int16_t nas_cellParams::srxlev

8.82 nas_CommInfo Struct Reference

Data Fields

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

8.82.1 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

8.82.2 Field Documentation

8.82.2.1 uint8_t nas_CommInfo::imsRegState

8.82.2.2 uint8_t nas_CommInfo::modemMode

8.82.2.3 uint8_t nas_CommInfo::psState

8.82.2.4 uint8_t nas_CommInfo::systemMode

8.82.2.5 int8_t nas_CommInfo::temperature

8.83 nas_CSGID Struct Reference

Data Fields

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

8.83.1 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

8.83.2 Field Documentation

8.83.2.1 uint32_t nas_CSGID::id

8.83.2.2 uint16_t nas_CSGID::mcc

8.83.2.3 uint16_t nas_CSGID::mnc

8.83.2.4 uint8_t nas_CSGID::mncPcsDigits

8.83.2.5 uint8_t nas_CSGID::rat

8.84 nas_currentPLMN Struct Reference

Data Fields

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

8.84.1 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

8.84.2 Field Documentation

8.84.2.1 uint16_t nas_currentPLMN::MCC

8.84.2.2 uint16_t nas_currentPLMN::MNC

8.84.2.3 uint8_t nas_currentPLMN::netDescr[255]

8.84.2.4 uint8_t nas_currentPLMN::netDescrLength

8.85 nas_dataSrvCapabilities Struct Reference

Data Fields

- uint8_t dataCapabilitiesLen
- uint8_t dataCapabilities [32]

8.85.1 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+

8.85.2 Field Documentation

8.85.2.1 uint8_t nas_dataSrvCapabilities::dataCapabilities[32]

8.85.2.2 uint8_t nas_dataSrvCapabilities::dataCapabilitiesLen

8.86 nas_DataStatusDetail Struct Reference

Data Fields

- uint32_t IPAddress
- uint8_t LastErrCode

8.86.1 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

8.86.2 Field Documentation

8.86.2.1 uint32_t nas_DataStatusDetail::IPAddress

8.86.2.2 uint8_t nas_DataStatusDetail::LastErrCode

8.87 nas_detailSvcInfo Struct Reference

Data Fields

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

8.87.1 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

8.87.2 Field Documentation

8.87.2.1 uint8_t nas_detailSvcInfo::hdrHybrid

8.87.2.2 uint8_t nas_detailSvcInfo::hdrSrvStatus

8.87.2.3 uint8_t nas_detailSvcInfo::isSysForbidden

8.87.2.4 uint8_t nas_detailSvcInfo::srvCapability

8.87.2.5 uint8_t nas_detailSvcInfo::srvStatus

8.88 nas_DeviceConfigDetail Struct Reference

Data Fields

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

8.88.1 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

8.88.2 Field Documentation

8.88.2.1 uint8_t nas_DeviceConfigDetail::Chipset

8.88.2.2 uint8_t nas_DeviceConfigDetail::HWVersion

8.88.2.3 uint8_t nas_DeviceConfigDetail::QLIC

8.88.2.4 uint8_t nas_DeviceConfigDetail::Technology

8.89 nas_dirNum Struct Reference

Data Fields

- uint8_t dirNumLen
- uint8_t dirNum [255]

8.89.1 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.

8.89.2 Field Documentation

8.89.2.1 uint8_t nas_dirNum::dirNum[255]

8.89.2.2 uint8_t nas_dirNum::dirNumLen

8.90 nas_DRCParams Struct Reference

Data Fields

- uint8_t DRCValue
- uint8_t DRCCover

8.90.1 Detailed Description

Parameters

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

8.90.2 Field Documentation

8.90.2.1 uint8_t nas_DRCParams::DRCCover

8.90.2.2 uint8_t nas_DRCParams::DRCValue

8.91 nas_ecioListElement Struct Reference

Data Fields

- int16_t ecio
- uint8_t radioIf

8.91.1 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

8.91.2 Field Documentation

8.91.2.1 int16_t nas_ecioListElement::ecio

8.91.2.2 uint8_t nas_ecioListElement::radioIf

8.92 nas_ECIOThresh Struct Reference

Data Fields

- uint8_t ECIOThresListLen
- int16_t * pECIOThresList

8.92.1 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

8.92.2 Field Documentation

8.92.2.1 uint8_t nas_ECIOThresh::ECIOThresListLen

8.92.2.2 int16_t* nas_ECIOThresh::pECIOThresList

8.93 nas_errorRateListElement Struct Reference

Data Fields

- uint16_t errorRate
- uint8_t radioIf

8.93.1 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

8.93.2 Field Documentation

8.93.2.1 uint16_t nas_errorRateListElement::errorRate

8.93.2.2 uint8_t nas_errorRateListElement::radioIf

8.94 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]

8.94.1 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.

8.94.2 Field Documentation

8.94.2.1 uint16_t nas_GERANInfo::arfcn

8.94.2.2 uint8_t nas_GERANInfo::bsic

8.94.2.3 uint32_t nas_GERANInfo::cellID

8.94.2.4 nas_nmrCellInfo nas_GERANInfo::insNmrCellInfo[255]

8.94.2.5 uint16_t nas_GERANInfo::lac

8.94.2.6 uint8_t nas_GERANInfo::nmrInst

8.94.2.7 uint8_t nas_GERANInfo::plmn[3]

8.94.2.8 uint16_t nas_GERANInfo::rxLev

8.94.2.9 uint32_t nas_GERANInfo::timingAdvance

8.95 nas_geranInstInfo Struct Reference

Data Fields

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

8.95.1 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.

8.95.2 Field Documentation

8.95.2.1 uint16_t nas_geranInstInfo::geranArfcn

8.95.2.2 uint8_t nas_geranInstInfo::geranBsicBcc

8.95.2.3 uint8_t nas_geranInstInfo::geranBsicNcc

8.95.2.4 int16_t nas_geranInstInfo::geranRssi

8.96 nas_gsmCellInfo Struct Reference

Data Fields

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

8.96.1 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.

8.96.2 Field Documentation

8.96.2.1 uint16_t nas_gsmCellInfo::arfcn

8.96.2.2 uint8_t nas_gsmCellInfo::band1900

8.96.2.3 uint8_t nas_gsmCellInfo::bsicId

8.96.2.4 uint8_t nas_gsmCellInfo::cellIdValid

8.96.2.5 int16_t nas_gsmCellInfo::rssi

8.96.2.6 int16_t nas_gsmCellInfo::srxlev

8.97 nas_GSMRSSIThresh Struct Reference

Data Fields

- uint8_t GSMRSSIThreshListLen
- int16_t * pGSMRSSIThreshList

8.97.1 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)

8.97.2 Field Documentation

8.97.2.1 uint8_t nas_GSMRSSIThresh::GSMRSSIThreshListLen

8.97.2.2 int16_t* nas_GSMRSSIThresh::pGSMRSSIThreshList

8.98 nas_GSMSrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t trueSrvStatus
- uint8_t isPrefDataPath

8.98.1 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

8.98.2 Field Documentation

8.98.2.1 uint8_t nas_GSMSrvStatusInfo::isPrefDataPath

8.98.2.2 uint8_t nas_GSMSrvStatusInfo::srvStatus

8.98.2.3 uint8_t nas_GSMSrvStatusInfo::trueSrvStatus

8.99 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

8.99.1 Detailed Description

Structure for storing the GSM System Information.

Parameters

sysInfoGSM	• See 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

8.99.2 Field Documentation

8.99.2.1 uint32_t nas_GSMSysInfo::cellId

8.99.2.2 uint8_t nas_GSMSysInfo::cellIdValid

8.99.2.3 uint8_t nas_GSMSysInfo::dtmSupp

8.99.2.4 uint8_t nas_GSMSysInfo::dtmSuppValid

8.99.2.5 uint8_t nas_GSMSysInfo::egprsSupp

8.99.2.6 uint8_t nas_GSMSysInfo::egprsSuppValid

8.99.2.7 uint16_t nas_GSMSysInfo::lac

8.99.2.8 uint8_t nas_GSMSysInfo::lacValid

8.99.2.9 uint8_t nas_GSMSysInfo::MCC[3]

8.99.2.10 uint8_t nas_GSMSysInfo::MNC[3]

8.99.2.11 uint8_t nas_GSMSysInfo::networkIdValid

8.99.2.12 uint8_t nas_GSMSysInfo::regRejectInfoValid

8.99.2.13 uint8_t nas_GSMSysInfo::rejCause

8.99.2.14 uint8_t nas_GSMSysInfo::rejectSrvDomain

8.99.2.15 nas_sysInfoCommon nas_GSMSysInfo::sysInfoGSM

8.100 nas_HDRECIOThresh Struct Reference

Data Fields

- uint8_t HDRECIOThreshListLen
- int16_t * pHDRECIOThreshList

8.100.1 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).

8.100.2 Field Documentation

8.100.2.1 uint8_t nas_HDRECIOThresh::HDRECIOThreshListLen

8.100.2.2 int16_t* nas_HDRECIOThresh::pHDRECIOThreshList

8.101 nas_HDRIOThresh Struct Reference

Data Fields

- uint8_t HDRIOThreshListLen
- int16_t * pHDRIOThreshList

8.101.1 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).

8.101.2 Field Documentation

8.101.2.1 uint8_t nas_HDRIOThresh::HDRIOThreshListLen

8.101.2.2 int16_t* nas_HDRIOThresh::pHDRIOThreshList

8.102 nas_HDRRSSIThresh Struct Reference

Data Fields

- uint8_t HDRRSSIThreshListLen
- int16_t * pHDRRSSIThreshList

8.102.1 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).

8.102.2 Field Documentation

8.102.2.1 uint8_t nas_HDRRSSIThresh::HDRRSSIThreshListLen

8.102.2.2 int16_t* nas_HDRRSSIThresh::pHDRRSSIThreshList

8.103 nas_HDRSINRThresh Struct Reference

Data Fields

- uint8_t HDRSINRThresListLen
- uint8_t * pHDRSINRThresList

8.103.1 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

8.103.2 Field Documentation

8.103.2.1 uint8_t nas_HDRSINRThresh::HDRSINRThresListLen

8.103.2.2 uint8_t* nas_HDRSINRThresh::pHDRSINRThresList

8.104 nas_HDRSINRThreshold Struct Reference

Data Fields

- uint8_t HDRSINRThreshListLen
- uint16_t * pHDRSINRThreshList

8.104.1 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

8.104.2 Field Documentation

8.104.2.1 uint8_t nas_HDRSINRThreshold::HDRSINRThreshListLen

8.104.2.2 uint16_t* nas_HDRSINRThreshold::pHDRSINRThreshList

8.105 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]

8.105.1 Detailed Description

Structure for storing the HDR System Information.

Parameters

sysInfoHDR	• See 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.

8.105.2 Field Documentation

8.105.2.1 uint8_t nas_HDRSysInfo::hdrActiveProt

8.105.2.2 uint8_t nas_HDRSysInfo::hdrActiveProtValid

8.105.2.3 uint8_t nas_HDRSysInfo::hdrPersonality

8.105.2.4 uint8_t nas_HDRSysInfo::hdrPersonalityValid

8.105.2.5 uint8_t nas_HDRSysInfo::is856SysId[16]

8.105.2.6 uint8_t nas_HDRSysInfo::is856SysIdValid

8.105.2.7 uint8_t nas_HDRSysInfo::isSysPrlMatch

8.105.2.8 uint8_t nas_HDRSysInfo::isSysPrlMatchValid

8.105.2.9 nas_sysInfoCommon nas_HDRSysInfo::sysInfoHDR

8.106 nas_homeSIDNID Struct Reference

Data Fields

- uint8_t numInstances
- nas_sidNid SidNid [255]

8.106.1 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

8.106.2 Field Documentation

8.106.2.1 uint8_t nas_homeSIDNID::numInstances

8.106.2.2 nas_sidNid nas_homeSIDNID::SidNid[255]

8.107 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]

8.107.1 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.

8.107.2 Field Documentation

8.107.2.1 uint8_t nas_infoInterFreq::cell_resel_priority

8.107.2.2 nas_cellParams nas_infoInterFreq::cellInterFreqParams[255]

8.107.2.3 uint8_t nas_infoInterFreq::cells_len

8.107.2.4 uint16_t nas_infoInterFreq::earfcn

8.107.2.5 uint8_t nas_infoInterFreq::threshXHigh

8.107.2.6 uint8_t nas_infoInterFreq::threshXLow

8.108 nas_IOThresh Struct Reference

Data Fields

- uint8_t IOThresListLen
- int32_t * pIOThresList

8.108.1 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

8.108.2 Field Documentation

8.108.2.1 uint8_t nas_IOThresh::IOThresListLen

8.108.2.2 int32_t* nas_IOThresh::pIOThresList

8.109 nas_lteEARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn0
- uint32_t earfcn1

8.109.1 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.

8.109.2 Field Documentation

8.109.2.1 uint32_t nas_lteEARFCN::earfcn0

8.109.2.2 uint32_t nas_lteEARFCN::earfcn1

8.109.2.3 uint8_t nas_lteEARFCN::status

8.110 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]

8.110.1 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.

8.110.2 Field Documentation

8.110.2.1 uint8_t nas_lteGsmCellInfo::cellReselPriority

8.110.2.2 uint8_t nas_lteGsmCellInfo::cells_len

8.110.2.3 nas_gsmCellInfo nas_lteGsmCellInfo::GsmCellInfo[255]

8.110.2.4 uint8_t nas_lteGsmCellInfo::nccPermitted

8.110.2.5 uint8_t nas_lteGsmCellInfo::threshGsmHigh

8.110.2.6 uint8_t nas_lteGsmCellInfo::threshGsmLow

8.111 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

8.111.1 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

8.111.2 Field Documentation

8.111.2.1 uint8_t nas_LTEInfo::band

8.111.2.2 uint8_t nas_LTEInfo::bandwidth

8.111.2.3 uint8_t nas_LTEInfo::emmConnState

8.111.2.4 uint8_t nas_LTEInfo::emmState

8.111.2.5 uint8_t nas_LTEInfo::emmSubState

8.111.2.6 uint16_t nas_LTEInfo::RXChan

8.111.2.7 uint16_t nas_LTEInfo::TXChan

8.112 nas_LTEInfoInterfreq Struct Reference

Data Fields

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

8.112.1 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.

8.112.2 Field Documentation

8.112.2.1 uint8_t nas_LTEInfoInterfreq::freqsLen

8.112.2.2 nas_infoInterFreq nas_LTEInfoInterfreq::InfoInterfreq[255]

8.112.2.3 uint8_t nas_LTEInfoInterfreq::ueInIdle

8.113 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]

8.113.1 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.

8.113.2 Field Documentation

8.113.2.1 nas_cellParams nas_LTEInfoIntrafreq::CellParams[255]

8.113.2.2 uint8_t nas_LTEInfoIntrafreq::cellReselPriority

8.113.2.3 uint8_t nas_LTEInfoIntrafreq::cellsLen

8.113.2.4 uint16_t nas_LTEInfoIntrafreq::earfcn

8.113.2.5 uint32_t nas_LTEInfoIntrafreq::globalCellId

8.113.2.6 uint8_t nas_LTEInfoIntrafreq::plmn[3]

8.113.2.7 uint16_t nas_LTEInfoIntrafreq::servingCellId

8.113.2.8 uint8_t nas_LTEInfoIntrafreq::sIntraSearch

8.113.2.9 uint8_t nas_LTEInfoIntrafreq::sNonIntraSearch

8.113.2.10 uint16_t nas_LTEInfoIntrafreq::tac

8.113.2.11 uint8_t nas_LTEInfoIntrafreq::threshServingLow

8.113.2.12 uint8_t nas_LTEInfoIntrafreq::ueInIdle

8.114 nas_LTEInfoNeighboringGSM Struct Reference

Data Fields

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

8.114.1 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.

8.114.2 Field Documentation

8.114.2.1 uint8_t nas_LTEInfoNeighboringGSM::freqsLen

8.114.2.2 nas_lteGsmCellInfo nas_LTEInfoNeighboringGSM::LteGsmCellInfo[255]

8.114.2.3 uint8_t nas_LTEInfoNeighboringGSM::ueInIdle

8.115 nas_LTEInfoNeighboringWCDMA Struct Reference

Data Fields

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

8.115.1 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.

8.115.2 Field Documentation

8.115.2.1 uint8_t nas_LTEInfoNeighboringWCDMA::freqsLen

8.115.2.2 nas_lteWcdmaCellInfo nas_LTEInfoNeighboringWCDMA::LTEWCDMACellInfo[255]

8.115.2.3 uint8_t nas_LTEInfoNeighboringWCDMA::ueInIdle

8.116 nas_ltePCI Struct Reference

Data Fields

- uint8_t status
- uint32_t earfcn
- uint32_t pci

8.116.1 Detailed Description

Parameters

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

8.116.2 Field Documentation

8.116.2.1 uint32_t nas_ltePCI::earfcn

8.116.2.2 uint32_t nas_ltePCI::pci

8.116.2.3 uint8_t nas_ltePCI::status

8.117 nas_lteRsrpinformation Struct Reference

Data Fields

- int16_t rsrplevel

8.117.1 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).

8.117.2 Field Documentation

8.117.2.1 int16_t nas_lteRsrpinformation::rsrplevel

8.118 nas_LTERSRPThresh Struct Reference

Data Fields

- uint8_t LTERSRPThreshListLen
- int16_t * pLTERSRPThreshList

8.118.1 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).

8.118.2 Field Documentation

8.118.2.1 uint8_t nas_LTERSRPThresh::LTERSRPThreshListLen

8.118.2.2 int16_t* nas_LTERSRPThresh::pLTERSRPThreshList

8.119 nas_LTERSRQThresh Struct Reference

Data Fields

- uint8_t LTERSRQThreshListLen
- int16_t * pLTERSRQThreshList

8.119.1 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)

8.119.2 Field Documentation

8.119.2.1 uint8_t nas_LTERSRQThresh::LTERSRQThreshListLen

8.119.2.2 int16_t* nas_LTERSRQThresh::pLTERSRQThreshList

8.120 nas_LTERSSIThresh Struct Reference

Data Fields

- uint8_t LTERSSIThreshListLen
- int16_t * pLTERSSIThreshList

8.120.1 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)

8.120.2 Field Documentation

8.120.2.1 uint8_t nas_LTERSSIThresh::LTERSSIThreshListLen

8.120.2.2 int16_t* nas_LTERSSIThresh::pLTERSSIThreshList

8.121 nas_LTESigRptCfg Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.121.1 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

8.121.2 Field Documentation

8.121.2.1 uint8_t nas_LTESigRptCfg::avgPeriod

8.121.2.2 uint8_t nas_LTESigRptCfg::rptRate

8.122 nas_LTESigRptConfig Struct Reference

Data Fields

- uint8_t rptRate
- uint8_t avgPeriod

8.122.1 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

8.122.2 Field Documentation

8.122.2.1 uint8_t nas_LTESigRptConfig::avgPeriod

8.122.2.2 uint8_t nas_LTESigRptConfig::rptRate

8.123 nas_lteSnrinformation Struct Reference

Data Fields

- int16_t snrlevel

8.123.1 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.

8.123.2 Field Documentation

8.123.2.1 int16_t nas_lteSnrinformation::snrlevel

8.124 nas_LTESNRThresh Struct Reference

Data Fields

- uint8_t LTESNRThresListLen
- int16_t * pLTESNRThresList

8.124.1 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

8.124.2 Field Documentation

8.124.2.1 uint8_t nas_LTESNRThresh::LTESNRThresListLen

8.124.2.2 int16_t* nas_LTESNRThresh::pLTESNRThresList

8.125 nas_LTESNRThreshold Struct Reference

Data Fields

- uint8_t LTESNRThreshListLen
- int16_t * pLTESNRThreshList

8.125.1 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).

8.125.2 Field Documentation

8.125.2.1 uint8_t nas_LTESNRThreshold::LTESNRThreshListLen

8.125.2.2 int16_t* nas_LTESNRThreshold::pLTESNRThreshList

8.126 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

8.126.1 Detailed Description

Structure for storing the LTE System Information.

Parameters

sysInfoLTE	• See 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

8.126.2 Field Documentation

8.126.2.1 uint32_t nas_LTESysInfo::cellId

8.126.2.2 uint8_t nas_LTESysInfo::cellIdValid

8.126.2.3 uint16_t nas_LTESysInfo::lac

8.126.2.4 uint8_t nas_LTESysInfo::lacValid

8.126.2.5 uint8_t nas_LTESysInfo::MCC[3]

8.126.2.6 uint8_t nas_LTESysInfo::MNC[3]

8.126.2.7 uint8_t nas_LTESysInfo::networkIdValid

8.126.2.8 uint8_t nas_LTESysInfo::regRejectInfoValid

8.126.2.9 uint8_t nas_LTESysInfo::rejCause

8.126.2.10 uint8_t nas_LTESysInfo::rejectSrvDomain

8.126.2.11 nas_sysInfoCommon nas_LTESysInfo::sysInfoLTE

8.126.2.12 uint16_t nas_LTESysInfo::tac

8.126.2.13 uint8_t nas_LTESysInfo::tacValid

8.127 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]

8.127.1 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 wcdmaCellInfo for more information.

8.127.2 Field Documentation

8.127.2.1 uint8_t nas_lteWcdmaCellInfo::cellReselPriority

8.127.2.2 uint8_t nas_lteWcdmaCellInfo::cellsLen

8.127.2.3 uint16_t nas_lteWcdmaCellInfo::threshXhigh

8.127.2.4 uint16_t nas_lteWcdmaCellInfo::threshXlow

8.127.2.5 uint16_t nas_lteWcdmaCellInfo::uarfcn

8.127.2.6 nas_wcdmaCellInfo nas_lteWcdmaCellInfo::WCDMACellInfo[255]

8.128 nas_minBasedIMSI Struct Reference

Data Fields

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

- uint8_t imsiMS2 [3]

8.128.1 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

8.128.2 Field Documentation

8.128.2.1 uint16_t nas_minBasedIMSI::imsiM1112

8.128.2.2 uint8_t nas_minBasedIMSI::imsiMS1[7]

8.128.2.3 uint8_t nas_minBasedIMSI::imsiMS2[3]

8.128.2.4 uint8_t nas_minBasedIMSI::mccM[3]

8.129 nas_MNRInfo Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint32_t rat

8.129.1 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

8.129.2 Field Documentation

8.129.2.1 uint16_t nas_MNRInfo::mcc

8.129.2.2 uint16_t nas_MNRInfo::mnc

8.129.2.3 uint32_t nas_MNRInfo::rat

8.130 nas_namName Struct Reference

Data Fields

- uint8_t namNameLen
- uint8_t namName [12]

8.130.1 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.

8.130.2 Field Documentation

8.130.2.1 uint8_t nas_namName::namName[12]

8.130.2.2 uint8_t nas_namName::namNameLen

8.131 nas_netSelectionPref Struct Reference

Data Fields

- uint8_t netReg
- uint16_t mcc
- uint16_t mnc

8.131.1 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

8.131.2 Field Documentation

8.131.2.1 uint16_t nas_netSelectionPref::mcc

8.131.2.2 uint16_t nas_netSelectionPref::mnc

8.131.2.3 uint8_t nas_netSelectionPref::netReg

8.132 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

8.132.1 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

8.132.2 Field Documentation

8.132.2.1 uint8_t nas_NetworkStat1x::ActSetCnt

8.132.2.2 uint8_t nas_NetworkStat1x::NeighborSetCnt

8.132.2.3 nas_ActPilotPNElement* nas_NetworkStat1x::pActPilotPNElements

8.132.2.4 uint16_t* nas_NetworkStat1x::pNeighborSetPilotPN

8.132.2.5 uint16_t nas_NetworkStat1x::RX_EC_IO

8.132.2.6 uint32_t nas_NetworkStat1x::RX_PWR

8.132.2.7 uint16_t nas_NetworkStat1x::SO

8.132.2.8 uint8_t nas_NetworkStat1x::State

8.132.2.9 uint32_t nas_NetworkStat1x::TX_PWR

8.133 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

8.133.1 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)

8.133.2 Field Documentation

8.133.2.1 uint8_t nas_NetworkStatEVDO::MACIndex

8.133.2.2 uint16_t nas_NetworkStatEVDO::PER

8.133.2.3 uint16_t nas_NetworkStatEVDO::PilotEnergy

8.133.2.4 uint16_t* nas_NetworkStatEVDO::pSectorID

8.133.2.5 uint16_t nas_NetworkStatEVDO::RX_PWR

8.133.2.6 uint8_t nas_NetworkStatEVDO::SectorIDLen

8.133.2.7 uint8_t nas_NetworkStatEVDO::SNR

8.133.2.8 uint8_t nas_NetworkStatEVDO::State

8.134 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

8.134.1 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

8.134.2 Field Documentation

8.134.2.1 uint16_t nas_nmrCellInfo::nmrArfcn

8.134.2.2 uint8_t nas_nmrCellInfo::nmrBsic

8.134.2.3 uint32_t nas_nmrCellInfo::nmrCellID

8.134.2.4 uint16_t nas_nmrCellInfo::nmrLac

8.134.2.5 uint8_t nas_nmrCellInfo::nmrPlmn[3]

8.134.2.6 uint16_t nas_nmrCellInfo::nmrRxLev

8.135 nas_operatorNameString Struct Reference

Data Fields

- uint8_t PLMNName [255]

8.135.1 Detailed Description

Parameters

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

8.135.2 Field Documentation

8.135.2.1 uint8_t nas_operatorNameString::PLMNName[255]

8.136 nas_OperatorPLMNData Struct Reference

Data Fields

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

- uint16_t lac1
- uint16_t lac2
- uint8_t PLMNRecID

8.136.1 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.

8.136.2 Field Documentation

8.136.2.1 uint16_t nas_OperatorPLMNData::lac1

8.136.2.2 uint16_t nas_OperatorPLMNData::lac2

8.136.2.3 uint8_t nas_OperatorPLMNData::mcc[3]

8.136.2.4 uint8_t nas_OperatorPLMNData::mnc[3]

8.136.2.5 uint8_t nas_OperatorPLMNData::PLMNRecID

8.137 nas_operatorPLMNList Struct Reference

Data Fields

- uint16_t numInstance
- nas_OperatorPLMNData PLMNData [255]

8.137.1 Detailed Description

Parameters

numInstance	• Number of sets of the elements.

PLMNData	• Refer OperatorPLMNLData for details (Optional).

8.137.2 Field Documentation

8.137.2.1 uint16_t nas_operatorPLMNList::numInstance

8.137.2.2 nas_OperatorPLMNData nas_operatorPLMNList::PLMNData[255]

8.138 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

8.138.1 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.

8.138.2 Field Documentation

8.138.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggPcellInfo::dl_bw_value

8.138.2.2 uint16_t nas_PhyCaAggPcellInfo::freq

8.138.2.3 uint16_t nas_PhyCaAggPcellInfo::iLTEbandValue

8.138.2.4 uint16_t nas_PhyCaAggPcellInfo::pci

8.138.2.5 uint8_t nas_PhyCaAggPcellInfo::TlvPresent

8.139 nas_PhyCaAggScellDlBw Struct Reference

Data Fields

- NAS_LTE_CPHY_CA_BW_NRB_LITE dl_bw_value
- uint8_t TlvPresent

8.139.1 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.

8.139.2 Field Documentation

8.139.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellDlBw::dl_bw_value

8.139.2.2 uint8_t nas_PhyCaAggScellDlBw::TlvPresent

8.140 nas_PhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.140.1 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.

8.140.2 Field Documentation

8.140.2.1 uint8_t nas_PhyCaAggScellIndex::scell_idx

8.140.2.2 uint8_t nas_PhyCaAggScellIndex::TlvPresent

8.141 nas_PhyCaAggScellIndType Struct Reference

Data Fields

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

8.141.1 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.

8.141.2 Field Documentation

8.141.2.1 uint16_t nas_PhyCaAggScellIndType::freq

8.141.2.2 uint16_t nas_PhyCaAggScellIndType::pci

8.141.2.3 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellIndType::scell_state

8.141.2.4 uint8_t nas_PhyCaAggScellIndType::TlvPresent

8.142 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

8.142.1 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.

8.142.2 Field Documentation

8.142.2.1 NAS_LTE_CPHY_CA_BW_NRB_LITE nas_PhyCaAggScellInfo::dl_bw_value

8.142.2.2 uint16_t nas_PhyCaAggScellInfo::freq

8.142.2.3 uint16_t nas_PhyCaAggScellInfo::iLTEbandValue

8.142.2.4 uint16_t nas_PhyCaAggScellInfo::pci

8.142.2.5 NAS_LTE_CPHY_SCELL_STATE_LITE nas_PhyCaAggScellInfo::scell_state

8.142.2.6 uint8_t nas_PhyCaAggScellInfo::TlvPresent

8.143 nas_PilotSetData Struct Reference

Data Fields

- uint8_t NumPilots
- nas_PilotSetParams * pPilotSetInfo

8.143.1 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.

8.143.2 Field Documentation

8.143.2.1 uint8_t nas_PilotSetData::NumPilots

8.143.2.2 nas_PilotSetParams* nas_PilotSetData::pPilotSetInfo

8.144 nas_PilotSetParams Struct Reference

Data Fields

- uint32_t PilotType
- uint16_t PilotPN
- uint16_t PilotStrength

8.144.1 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)

8.144.2 Field Documentation

8.144.2.1 uint16_t nas_PilotSetParams::PilotPN

8.144.2.2 uint16_t nas_PilotSetParams::PilotStrength

8.144.2.3 uint32_t nas_PilotSetParams::PilotType

8.145 nas_PLMNNetworkName Struct Reference

Data Fields

- uint8_t numInstance
- nas_PLMNNetworkNameData PLMNNetName [255]

8.145.1 Detailed Description

Parameters

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

8.145.2 Field Documentation

8.145.2.1 uint8_t nas_PLMNNetworkName::numInstance

8.145.2.2 nas_PLMNNetworkNameData nas_PLMNNetworkName::PLMNNetName[255]

8.146 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]

8.146.1 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.

8.146.2 Field Documentation

8.146.2.1 uint8_t nas_PLMNNetworkNameData::codingScheme

8.146.2.2 uint8_t nas_PLMNNetworkNameData::countryInitials

8.146.2.3 uint8_t nas_PLMNNetworkNameData::longName[255]

8.146.2.4 uint8_t nas_PLMNNetworkNameData::longNameLen

8.146.2.5 uint8_t nas_PLMNNetworkNameData::longNameSpareBits

8.146.2.6 uint8_t nas_PLMNNetworkNameData::shortName[255]

8.146.2.7 uint8_t nas_PLMNNetworkNameData::shortNameLen

8.146.2.8 uint8_t nas_PLMNNetworkNameData::shortNameSpareBits

8.147 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

8.147.1 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

8.147.2 Field Documentation

8.147.2.1 uint16_t nas_protocolSubtypeElement::AccessMac

8.147.2.2 uint16_t nas_protocolSubtypeElement::AuthProt

8.147.2.3 uint16_t nas_protocolSubtypeElement::ControlMac

8.147.2.4 uint16_t nas_protocolSubtypeElement::EncryptProt

8.147.2.5 uint16_t nas_protocolSubtypeElement::ForwardMac

8.147.2.6 uint16_t nas_protocolSubtypeElement::IdleState

8.147.2.7 uint16_t nas_protocolSubtypeElement::KeyExchange

8.147.2.8 uint16_t nas_protocolSubtypeElement::MultDisc

8.147.2.9 uint16_t nas_protocolSubtypeElement::PhysicalLayer

8.147.2.10 uint16_t nas_protocolSubtypeElement::ReverseMac

8.147.2.11 uint16_t nas_protocolSubtypeElement::SecProt

8.147.2.12 uint16_t nas_protocolSubtypeElement::VirtStream

8.148 nas_qaQmi3Gpp2TimeZone Struct Reference

Data Fields

- uint8_t leapSeconds
- uint8_t localTimeOffset
- uint8_t daylightSavings

8.148.1 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)

8.148.2 Field Documentation

8.148.2.1 uint8_t nas_qaQmi3Gpp2TimeZone::daylightSavings

8.148.2.2 uint8_t nas_qaQmi3Gpp2TimeZone::leapSeconds

8.148.2.3 uint8_t nas_qaQmi3Gpp2TimeZone::localTimeOffset

8.149 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]

8.149.1 Detailed Description

Parameters

MCC	Mobile Country Code
MNC	Mobile Network Code
InUse	current network or not
Roaming	Home/Roam Status of the Network
Forbidden	Network in the forbidden network list or not
Preferred	Network in the Preferred network list or not
Desription	Network Name/Description

8.149.2 Field Documentation

8.149.2.1 char nas_QmiNas3GppNetworkInfo::Desription[255]

8.149.2.2 uint32_t nas_QmiNas3GppNetworkInfo::Forbidden

8.149.2.3 uint32_t nas_QmiNas3GppNetworkInfo::InUse

8.149.2.4 uint16_t nas_QmiNas3GppNetworkInfo::MCC

8.149.2.5 uint16_t nas_QmiNas3GppNetworkInfo::MNC

8.149.2.6 uint32_t nas_QmiNas3GppNetworkInfo::Preferred

8.149.2.7 uint32_t nas_QmiNas3GppNetworkInfo::Roaming

8.150 nas_QmiNas3GppNetworkRAT Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t RAT

8.150.1 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

8.150.2 Field Documentation

8.150.2.1 uint16_t nas_QmiNas3GppNetworkRAT::MCC

8.150.2.2 uint16_t nas_QmiNas3GppNetworkRAT::MNC

8.150.2.3 uint8_t nas_QmiNas3GppNetworkRAT::RAT

8.151 nas_QmisNasPcsDigit Struct Reference

Data Fields

- uint16_t MCC
- uint16_t MNC
- uint8_t includes_pcs_digit

8.151.1 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

8.151.2 Field Documentation

8.151.2.1 uint8_t nas_QmisNasPcsDigit::includes_pcs_digit

8.151.2.2 uint16_t nas_QmisNasPcsDigit::MCC

8.151.2.3 uint16_t nas_QmisNasPcsDigit::MNC

8.152 nas_RejectReasonTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t serviceDomain
- uint32_t rejectCause

8.152.1 Detailed Description

Parameters

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

8.152.2 Field Documentation

8.152.2.1 uint32_t nas_RejectReasonTlv::rejectCause

8.152.2.2 uint32_t nas_RejectReasonTlv::serviceDomain

8.152.2.3 uint8_t nas_RejectReasonTlv::TlvPresent

8.153 nas_RFInfoTlv Struct Reference

Data Fields

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

8.153.1 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

8.153.2 Field Documentation

8.153.2.1 uint32_t nas_RFInfoTlv::activeBandClass[255]

8.153.2.2 uint32_t nas_RFInfoTlv::activeChannel[255]

8.153.2.3 uint32_t nas_RFInfoTlv::radioInterface[255]

8.153.2.4 uint8_t nas_RFInfoTlv::radioInterfaceSize

8.153.2.5 uint8_t nas_RFInfoTlv::TlvPresent

8.154 nas_roamIndList Struct Reference

Data Fields

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

8.154.1 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

8.154.2 Field Documentation

8.154.2.1 uint8_t nas_roamIndList::numInstances

8.154.2.2 uint8_t nas_roamIndList::radioInterface[32]

8.154.2.3 uint8_t nas_roamIndList::roamIndicator[32]

8.155 nas_RSRPThresh Struct Reference

Data Fields

- uint8_t RSRPThresListLen
- int16_t * pRSRPThresList

8.155.1 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

8.155.2 Field Documentation

8.155.2.1 int16_t* nas_RSRPThresh::pRSRPThresList

8.155.2.2 uint8_t nas_RSRPThresh::RSRPThresListLen

8.156 nas_rsrqInformation Struct Reference

Data Fields

- int8_t rsrq
- uint8_t radioIf

8.156.1 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

8.156.2 Field Documentation

8.156.2.1 uint8_t nas_rsrqInformation::radioIf

8.156.2.2 int8_t nas_rsrqInformation::rsrq

8.157 nas_RSRQThresh Struct Reference

Data Fields

- uint8_t RSRQThresListLen
- int8_t * pRSRQThresList

8.157.1 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

8.157.2 Field Documentation

8.157.2.1 int8_t* nas_RSRQThresh::pRSRQThresList

8.157.2.2 uint8_t nas_RSRQThresh::RSRQThresListLen

8.158 nas_RSSIThresh Struct Reference

Data Fields

- uint8_t RSSIThresListLen
- int8_t * pRSSIThresList

8.158.1 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.

8.158.2 Field Documentation

8.158.2.1 int8_t* nas_RSSIThresh::pRSSIThresList

8.158.2.2 uint8_t nas_RSSIThresh::RSSIThresListLen

8.159 nas_rxInfo Struct Reference

Data Fields

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

8.159.1 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.

8.159.2 Field Documentation

8.159.2.1 int32_t nas_rxInfo::ecio

8.159.2.2 uint8_t nas_rxInfo::isRadioTuned

8.159.2.3 uint32_t nas_rxInfo::phase

8.159.2.4 int32_t nas_rxInfo::rscp

8.159.2.5 int32_t nas_rxInfo::rsrp

8.159.2.6 int32_t nas_rxInfo::rxPower

8.160 nas_RxSigInfo Struct Reference

Data Fields

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

8.160.1 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

8.160.2 Field Documentation

8.160.2.1 uint8_t nas_RxSigInfo::isRadioTuned

8.160.2.2 int32_t nas_RxSigInfo::rsrp

8.160.2.3 uint8_t nas_RxSigInfo::rxChainIndex

8.160.2.4 int32_t nas_RxSigInfo::rxPower

8.161 nas_rxSignalStrengthListElement Struct Reference

Data Fields

- int16_t rxSignalStrength
- uint8_t radioIf

8.161.1 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.

8.161.2 Field Documentation

8.161.2.1 uint8_t nas_rxSignalStrengthListElement::radioIf

8.161.2.2 int16_t nas_rxSignalStrengthListElement::rxSignalStrength

8.162 nas_SccRxInfo Struct Reference

Data Fields

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

8.162.1 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.

8.162.2 Field Documentation

8.162.2.1 uint8_t nas_SccRxInfo::numInstances

8.162.2.2 int32_t nas_SccRxInfo::rsrq

8.162.2.3 nas_RxSigInfo nas_SccRxInfo::sigInfo[255]

8.162.2.4 int16_t nas_SccRxInfo::snr

8.162.2.5 uint8_t nas_SccRxInfo::TlvPresent

8.163 nas_serviceProviderName Struct Reference

Data Fields

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

8.163.1 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.

8.163.2 Field Documentation

8.163.2.1 uint8_t nas_serviceProviderName::displayCondition

8.163.2.2 uint8_t nas_serviceProviderName::spn[255]

8.163.2.3 uint8_t nas_serviceProviderName::spnLength

8.164 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]

8.164.1 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

8.164.2 Field Documentation

8.164.2.1 uint8_t nas_servSystem::csAttachState

8.164.2.2 uint8_t nas_servSystem::numRadioInterfaces

8.164.2.3 uint8_t nas_servSystem::psAttachState

8.164.2.4 uint8_t nas_servSystem::radioInterface[32]

8.164.2.5 uint8_t nas_servSystem::regState

8.164.2.6 uint8_t nas_servSystem::selNetwork

8.165 nas_sidNid Struct Reference

Data Fields

- uint16_t nid
- uint16_t sid

8.165.1 Detailed Description

This structure contains the parameters for SidNid Information

Parameters

nid	• Network ID
sid	• System ID

8.165.2 Field Documentation

8.165.2.1 uint16_t nas_sidNid::nid

8.165.2.2 uint16_t nas_sidNid::sid

8.166 nas_SignalStrengthTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- int8_t signalStrength
- uint32_t radioInterface

8.166.1 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

8.166.2 Field Documentation

8.166.2.1 uint32_t nas_SignalStrengthTlv::radioInterface

8.166.2.2 int8_t nas_SignalStrengthTlv::signalStrength

8.166.2.3 uint8_t nas_SignalStrengthTlv::TlvPresent

8.167 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

8.167.1 Detailed Description

Parameters

rxSignal-StrengthDelta	RSSI delta(in dBm) at which an event report indication
ecioDelta	ecio delta
ioDelta	io delta
sinrDelta	sinr delta
rsrqDelta	rsrq delta
ecioThreshold-ListLen	
ecioThreshold-List	
sinrThreshold-ListLen	
sinrThreshold-List	
lteSnrDelta	lte snr delta
lteRsrpDelta	lte rsrp delta

8.167.2 Field Documentation

8.167.2.1 uint8_t nas_SLQSSignalStrengthsIndReq::ecioDelta

8.167.2.2 int16_t nas_SLQSSignalStrengthsIndReq::ecioThresholdList[10]

8.167.2.3 uint8_t nas_SLQSSignalStrengthsIndReq::ecioThresholdListLen

8.167.2.4 uint8_t nas_SLQSSignalStrengthsIndReq::ioDelta

8.167.2.5 uint8_t nas_SLQSSignalStrengthsIndReq::lteRsrpDelta

8.167.2.6 uint16_t nas_SLQSSignalStrengthsIndReq::lteSnrDelta

8.167.2.7 uint8_t nas_SLQSSignalStrengthsIndReq::rsrqDelta

8.167.2.8 uint8_t nas_SLQSSignalStrengthsIndReq::rxSignalStrengthDelta

8.167.2.9 uint8_t nas_SLQSSignalStrengthsIndReq::sinrDelta

8.167.2.10 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdList[5]

8.167.2.11 uint8_t nas_SLQSSignalStrengthsIndReq::sinrThresholdListLen

8.168 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

8.168.1 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

8.168.2 Field Documentation

8.168.2.1 nas_ecioListElement nas_SLQSSignalStrengthsInformation::ecioInfo

8.168.2.2 nas_errorRateListElement nas_SLQSSignalStrengthsInformation::errorRateInfo

8.168.2.3 uint32_t nas_SLQSSignalStrengthsInformation::io

8.168.2.4 nas_lteRsrpinformation nas_SLQSSignalStrengthsInformation::lteRsrpinfo

8.168.2.5 nas_lteSnrinformation nas_SLQSSignalStrengthsInformation::lteSnrinfo

8.168.2.6 nas_rsrqInformation nas_SLQSSignalStrengthsInformation::rsrqInfo

8.168.2.7 nas_rxSignalStrengthListElement nas_SLQSSignalStrengthsInformation::rxSignalStrengthInfo

8.168.2.8 uint8_t nas_SLQSSignalStrengthsInformation::sinr

8.169 nas_SLQSSignalStrengthsTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- nas_SLQSSignalStrengthsInformation sSLQSSignalStrengthsInfo

8.169.1 Detailed Description

Parameters

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

8.169.2 Field Documentation

8.169.2.1 nas_SLQSSignalStrengthsInformation nas_SLQSSignalStrengthsTlv::sSLQSSignalStrengthsInfo

8.169.2.2 uint8_t nas_SLQSSignalStrengthsTlv::TlvPresent

8.170 nas_SrvStatusInfo Struct Reference

Data Fields

- uint8_t srvStatus
- uint8_t isPrefDataPath

8.170.1 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

8.170.2 Field Documentation

8.170.2.1 uint8_t nas_SrvStatusInfo::isPrefDataPath

8.170.2.2 uint8_t nas_SrvStatusInfo::srvStatus

8.171 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

8.171.1 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

8.171.2 Field Documentation

8.171.2.1 uint8_t nas_sysInfoCommon::isSysForbidden

8.171.2.2 uint8_t nas_sysInfoCommon::isSysForbiddenValid

8.171.2.3 uint8_t nas_sysInfoCommon::roamStatus

8.171.2.4 uint8_t nas_sysInfoCommon::roamStatusValid

8.171.2.5 uint8_t nas_sysInfoCommon::srvCapability

8.171.2.6 uint8_t nas_sysInfoCommon::srvCapabilityValid

8.171.2.7 uint8_t nas_sysInfoCommon::srvDomain

8.171.2.8 uint8_t nas_sysInfoCommon::srvDomainValid

8.172 nas_TDSCDMAECIOThresh Struct Reference

Data Fields

- uint8_t TDSCDMAECIOThreshListLen
- float * pTDSCDMAECIOThreshList

8.172.1 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.

8.172.2 Field Documentation

8.172.2.1 float* nas_TDSCDMAECIOThresh::pTDSCDMAECIOThreshList

8.172.2.2 uint8_t nas_TDSCDMAECIOThresh::TDSCDMAECIOThreshListLen

8.173 nas_TDSCDMARSCPThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSCPThreshListLen
- int16_t * pTDSCDMARSCPThreshList

8.173.1 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).

8.173.2 Field Documentation

8.173.2.1 int16_t* nas_TDSCDMARSCPThresh::pTDSCDMARSCPThreshList

8.173.2.2 uint8_t nas_TDSCDMARSCPThresh::TDSCDMARSCPThreshListLen

8.174 nas_TDSCDMARSSIThresh Struct Reference

Data Fields

- uint8_t TDSCDMARSSIThreshListLen
- float * pTDSCDMARSSIThreshList

8.174.1 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.

8.174.2 Field Documentation

8.174.2.1 float* nas_TDSCDMARSSIThresh::pTDSCDMARSSIThreshList

8.174.2.2 uint8_t nas_TDSCDMARSSIThresh::TDSCDMARSSIThreshListLen

8.175 nas_TDSCDMASINRCONFThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRCONFThreshListLen
- float * pTDSCDMASINRCONFThreshList

8.175.1 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

8.175.2 Field Documentation

8.175.2.1 float* nas_TDSCDMASINRCONFThresh::pTDSCDMASINRCONFThreshList

8.175.2.2 uint8_t nas_TDSCDMASINRCONFThresh::TDSCDMASINRCONFThreshListLen

8.176 nas_TDSCDMASINRThresh Struct Reference

Data Fields

- uint8_t TDSCDMASINRThreshListLen
- float * pTDSCDMASINRThreshList

8.176.1 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

8.176.2 Field Documentation

8.176.2.1 float* nas_TDSCDMASINRThresh::pTDSCDMASINRThreshList

8.176.2.2 uint8_t nas_TDSCDMASINRThresh::TDSCDMASINRThreshListLen

8.177 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

8.177.1 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.

8.177.2 Field Documentation

8.177.2.1 uint8_t nas_timeInfo::day

8.177.2.2 uint8_t nas_timeInfo::dayLtSavingAdj

8.177.2.3 uint8_t nas_timeInfo::dayOfWeek

8.177.2.4 uint8_t nas_timeInfo::hour

8.177.2.5 uint8_t nas_timeInfo::minute

8.177.2.6 uint8_t nas_timeInfo::month

8.177.2.7 uint8_t nas_timeInfo::radioInterface

8.177.2.8 uint8_t nas_timeInfo::second

8.177.2.9 int8_t nas_timeInfo::timeZone

8.177.2.10 uint8_t nas_timeInfo::TlvPresent

8.177.2.11 uint16_t nas_timeInfo::year

8.178 nas_trueIMSI Struct Reference

Data Fields

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

8.178.1 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

8.178.2 Field Documentation

8.178.2.1 uint16_t nas_trueIMSI::imsiT1112

8.178.2.2 uint8_t nas_trueIMSI::imsiTaddrNum

8.178.2.3 uint8_t nas_trueIMSI::imsiTS1[7]

8.178.2.4 uint8_t nas_trueIMSI::imsiTS2[3]

8.178.2.5 uint8_t nas_trueIMSI::mccT[3]

8.179 nas_txInfo Struct Reference

Data Fields

- uint8_t isInTraffic
- int32_t txPower

8.179.1 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.

8.179.2 Field Documentation

8.179.2.1 uint8_t nas_txInfo::isInTraffic

8.179.2.2 int32_t nas_txInfo::txPower

8.180 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]

8.180.1 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.

8.180.2 Field Documentation

8.180.2.1 uint16_t nas_UMTSInfo::cellID

8.180.2.2 int16_t nas_UMTSInfo::ecio

8.180.2.3 uint8_t nas_UMTSInfo::geranInst

8.180.2.4 nas_geranInstInfo nas_UMTSInfo::GeranInstInfo[255]

8.180.2.5 uint16_t nas_UMTSInfo::lac

8.180.2.6 uint8_t nas_UMTSInfo::plmn[3]

8.180.2.7 uint16_t nas_UMTSInfo::psc

8.180.2.8 int16_t nas_UMTSInfo::rscp

8.180.2.9 uint16_t nas_UMTSInfo::uarfcn

8.180.2.10 uint8_t nas_UMTSInfo::umtsInst

8.180.2.11 nas_UMTSinstInfo nas_UMTSInfo::UMTSInstInfo[255]

8.181 nas_UMTSinstInfo Struct Reference

Data Fields

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

8.181.1 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).

8.181.2 Field Documentation

8.181.2.1 int16_t nas_UMTSinstInfo::umtsEcio

8.181.2.2 uint16_t nas_UMTSinstInfo::umtsPsc

8.181.2.3 int16_t nas_UMTSinstInfo::umtsRscp

8.181.2.4 uint16_t nas_UMTSinstInfo::umtsUarfcn

8.182 nas_umtsLTENbrCell Struct Reference

Data Fields

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

8.182.1 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.

8.182.2 Field Documentation

8.182.2.1 uint8_t nas_umtsLTENbrCell::cellIsTDD

8.182.2.2 uint16_t nas_umtsLTENbrCell::earfcn

8.182.2.3 uint16_t nas_umtsLTENbrCell::pci

8.182.2.4 uint32_t nas_umtsLTENbrCell::rsrp

8.182.2.5 uint32_t nas_umtsLTENbrCell::rsrq

8.182.2.6 int16_t nas_umtsLTENbrCell::srxlev

8.183 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

8.183.1 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.

8.183.2 Field Documentation

8.183.2.1 uint8_t nas_UniversalTime::day

8.183.2.2 uint8_t nas_UniversalTime::dayOfWeek

8.183.2.3 uint8_t nas_UniversalTime::hour

8.183.2.4 uint8_t nas_UniversalTime::minute

8.183.2.5 uint8_t nas_UniversalTime::month

8.183.2.6 uint8_t nas_UniversalTime::second

8.183.2.7 uint16_t nas_UniversalTime::year

8.184 nas_wcdmaCellInfo Struct Reference

Data Fields

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

8.184.1 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.

8.184.2 Field Documentation

8.184.2.1 int16_t nas_wcdmaCellInfo::cpich_ecno

8.184.2.2 int16_t nas_wcdmaCellInfo::cpich_rscp

8.184.2.3 uint16_t nas_wcdmaCellInfo::psc

8.184.2.4 int16_t nas_wcdmaCellInfo::srxlev

8.185 nas_WCDMAECIOThresh Struct Reference

Data Fields

- uint8_t WCDMAECIOThreshListLen
- int16_t * pWCDMAECIOThreshList

8.185.1 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)

8.185.2 Field Documentation

8.185.2.1 int16_t* nas_WCDMAECIOThresh::pWCDMAECIOThreshList

8.185.2.2 uint8_t nas_WCDMAECIOThresh::WCDMAECIOThreshListLen

8.186 nas_WCDMAInfoLTENeighborCell Struct Reference

Data Fields

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

8.186.1 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.

8.186.2 Field Documentation

8.186.2.1 nas_umtsLTENbrCell nas_WCDMAInfoLTENeighborCell::UMTSLTENbrCell[255]

8.186.2.2 uint8_t nas_WCDMAInfoLTENeighborCell::umtsLTENbrCellLen

8.186.2.3 uint32_t nas_WCDMAInfoLTENeighborCell::wcdmaRRCState

8.187 nas_WCDMARSSIThresh Struct Reference

Data Fields

- uint8_t WCDMARSSIThreshListLen
- int16_t * pWCDMARSSIThreshList

8.187.1 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)

8.187.2 Field Documentation

8.187.2.1 int16_t* nas_WCDMARSSIThresh::pWCDMARSSIThreshList

8.187.2.2 uint8_t nas_WCDMARSSIThresh::WCDMARSSIThreshListLen

8.188 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

8.188.1 Detailed Description

Structure for storing the WCDMA System Information.

Parameters

sysInfoWCDMA	• See 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.
hsCallStatusValid	• 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

8.188.2 Field Documentation

8.188.2.1 uint32_t nas_WCDMASysInfo::cellId

8.188.2.2 uint8_t nas_WCDMASysInfo::cellIdValid

8.188.2.3 uint8_t nas_WCDMASysInfo::hsCallStatus

8.188.2.4 uint8_t nas_WCDMASysInfo::hsCallStatusValid

8.188.2.5 uint8_t nas_WCDMASysInfo::hsInd

8.188.2.6 uint8_t nas_WCDMASysInfo::hsIndValid

8.188.2.7 uint16_t nas_WCDMASysInfo::lac

8.188.2.8 uint8_t nas_WCDMASysInfo::lacValid

8.188.2.9 uint8_t nas_WCDMASysInfo::MCC[3]

8.188.2.10 uint8_t nas_WCDMASysInfo::MNC[3]

8.188.2.11 uint8_t nas_WCDMASysInfo::networkIdValid

8.188.2.12 uint16_t nas_WCDMASysInfo::psc

8.188.2.13 uint8_t nas_WCDMASysInfo::pscValid

8.188.2.14 uint8_t nas_WCDMASysInfo::regRejectInfoValid

8.188.2.15 uint8_t nas_WCDMASysInfo::rejCause

8.188.2.16 uint8_t nas_WCDMASysInfo::rejectSrvDomain

8.188.2.17 nas_sysInfoCommon nas_WCDMASysInfo::sysInfoWCDMA

8.189 nas_wcdmaUARFCN Struct Reference

Data Fields

- uint8_t status
- uint32_t uarfcn

8.189.1 Detailed Description

Parameters

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

8.189.2 Field Documentation

8.189.2.1 uint8_t nas_wcdmaUARFCN::status

8.189.2.2 uint32_t nas_wcdmaUARFCN::uarfcn

8.190 NASBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t band_pref

8.190.1 Field Documentation

8.190.1.1 uint64_t NASBandPreferenceTlv::band_pref

8.190.1.2 uint8_t NASBandPreferenceTlv::TlvPresent

8.191 NASEmergencyModeTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t EmerMode

8.191.1 Field Documentation

8.191.1.1 uint8_t NASEmergencyModeTlv::EmerMode

8.191.1.2 uint8_t NASEmergencyModeTlv::TlvPresent

8.192 NasGetLTECphyCaInfo Struct Reference

Data Fields

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

8.192.1 Field Documentation

8.192.1.1 NASPhyCaAggPcellInfo NasGetLTECphyCaInfo::PhyCaAggPcellInfo

8.192.1.2 NASPhyCaAggScellDlBw NasGetLTECphyCaInfo::PhyCaAggScellDlBw

8.192.1.3 NASPhyCaAggScellIndex NasGetLTECphyCaInfo::PhyCaAggScellIndex

8.192.1.4 NASPhyCaAggScellIndType NasGetLTECphyCaInfo::PhyCaAggScellIndType

8.192.1.5 NASPhyCaAggScellInfo NasGetLTECphyCaInfo::PhyCaAggScellInfo

8.193 NASGWAcqOrderPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t GWAcqOrderPref

8.193.1 Field Documentation

8.193.1.1 uint32_t NASGWAcqOrderPrefTlv::GWAcqOrderPref

8.193.1.2 uint8_t NASGWAcqOrderPrefTlv::TlvPresent

8.194 NASLTEBandPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t LTEBandPref

8.194.1 Field Documentation

8.194.1.1 uint64_t NASLTEBandPreferenceTlv::LTEBandPref

8.194.1.2 uint8_t NASLTEBandPreferenceTlv::TlvPresent

8.195 NASLteNasReleaseInfoTlv Struct Reference

Data Fields

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

8.195.1 Field Documentation

8.195.1.1 uint8_t NASLteNasReleaseInfoTlv::nas_major

8.195.1.2 uint8_t NASLteNasReleaseInfoTlv::nas_minor

8.195.1.3 uint8_t NASLteNasReleaseInfoTlv::nas_release

8.195.1.4 uint8_t NASLteNasReleaseInfoTlv::TlvPresent

8.196 NASModePreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t ModePref

8.196.1 Field Documentation

8.196.1.1 uint16_t NASModePreferenceTlv::ModePref

8.196.1.2 uint8_t NASModePreferenceTlv::TlvPresent

8.197 NASNetSelPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t NetSelPref

8.197.1 Field Documentation

8.197.1.1 uint8_t NASNetSelPreferenceTlv::NetSelPref

8.197.1.2 uint8_t NASNetSelPreferenceTlv::TlvPresent

8.198 NASOTAMessageTlv Struct Reference

Data Fields

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

8.198.1 Field Documentation

8.198.1.1 uint8_t NASOTAMessageTlv::data_buf[2048]

8.198.1.2 uint16_t NASOTAMessageTlv::data_len

8.198.1.3 uint32_t NASOTAMessageTlv::message_type

8.198.1.4 uint8_t NASOTAMessageTlv::TlvPresent

8.199 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

8.199.1 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 for more information.
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE for more information.
TlvPresent	• Tlv Present.

8.199.2 Field Documentation

8.199.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggPcellInfo::dl_bw_value

8.199.2.2 uint32_t NASPhyCaAggPcellInfo::freq

8.199.2.3 uint32_t NASPhyCaAggPcellInfo::iLTEbandValue

8.199.2.4 uint32_t NASPhyCaAggPcellInfo::pci

8.199.2.5 uint8_t NASPhyCaAggPcellInfo::TlvPresent

8.200 NASPhyCaAggScellDlBw Struct Reference

Data Fields

- LITEQMI_NAS_LTE_CPHY_CA_BW_NRB dl_bw_value
- uint8_t TlvPresent

8.200.1 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 for more information.

8.200.2 Field Documentation

8.200.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellDlBw::dl_bw_value

8.200.2.2 uint8_t NASPhyCaAggScellDlBw::TlvPresent

8.201 NASPhyCaAggScellIndex Struct Reference

Data Fields

- uint8_t scell_idx
- uint8_t TlvPresent

8.201.1 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.

8.201.2 Field Documentation

8.201.2.1 uint8_t NASPhyCaAggScellIndex::scell_idx

8.201.2.2 uint8_t NASPhyCaAggScellIndex::TlvPresent

8.202 NASPhyCaAggScellIndType Struct Reference

Data Fields

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

8.202.1 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 NAS_LTE_CPHY_SCELL_STATE for more information.
TlvPresent	• Tlv Present.

8.202.2 Field Documentation

8.202.2.1 uint32_t NASPhyCaAggScellIndType::freq

8.202.2.2 uint32_t NASPhyCaAggScellIndType::pci

8.202.2.3 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellIndType::scell_state

8.202.2.4 uint8_t NASPhyCaAggScellIndType::TlvPresent

8.203 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

8.203.1 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 for more information.
iLTEbandValue	• Band value. • Range for LTE Band class 120 to 160.
scell_state	• Scell state Values. • See NAS_LTE_CPHY_SCELL_STATE for more information.
TlvPresent	• Tlv Present.

8.203.2 Field Documentation

8.203.2.1 LITEQMI_NAS_LTE_CPHY_CA_BW_NRB NASPhyCaAggScellInfo::dl_bw_value

8.203.2.2 uint32_t NASPhyCaAggScellInfo::freq

8.203.2.3 uint32_t NASPhyCaAggScellInfo::iLTEbandValue

8.203.2.4 uint32_t NASPhyCaAggScellInfo::pci

8.203.2.5 LITEQMI_NAS_LTE_CPHY_SCELL_STATE NASPhyCaAggScellInfo::scell_state

8.203.2.6 uint8_t NASPhyCaAggScellInfo::TlvPresent

8.204 NASPRLPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t PRLPref

8.204.1 Field Documentation

8.204.1.1 uint16_t NASPRLPreferenceTlv::PRLPref

8.204.1.2 uint8_t NASPRLPreferenceTlv::TlvPresent

8.205 NASQmiCbkNasSwiOTAMessageInd Struct Reference

Data Fields

- NASOTAMessageTlv otaMsgTlv
- NASLteNasReleaseInfoTlv nasRelInfoTlv
- NASTimeInfoTlv timeTlv

8.205.1 Field Documentation

8.205.1.1 NASLteNasReleaseInfoTlv NASQmiCbkNasSwiOTAMessageInd::nasRelInfoTlv

8.205.1.2 NASOTAMessageTlv NASQmiCbkNasSwiOTAMessageInd::otaMsgTlv

8.205.1.3 NASTimeInfoTlv NASQmiCbkNasSwiOTAMessageInd::timeTlv

8.206 NASQmiCbkNasSystemSelPrefInd Struct Reference

Data Fields

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

8.206.1 Field Documentation

8.206.1.1 NASBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::BPTlv

8.206.1.2 NASEmergencyModeTlv NASQmiCbkNasSystemSelPrefInd::EMTlv

8.206.1.3 NASGWAcqOrderPrefTlv NASQmiCbkNasSystemSelPrefInd::GWAOPTlv

8.206.1.4 NASLTEBandPreferenceTlv NASQmiCbkNasSystemSelPrefInd::LBPTlv

8.206.1.5 NASModePreferenceTlv NASQmiCbkNasSystemSelPrefInd::MPTlv

8.206.1.6 NASNetSelPreferenceTlv NASQmiCbkNasSystemSelPrefInd::NSPTlv

8.206.1.7 NASPRLPreferenceTlv NASQmiCbkNasSystemSelPrefInd::PRLPTlv

8.206.1.8 NASRoamPreferenceTlv NASQmiCbkNasSystemSelPrefInd::RPTlv

8.206.1.9 NASServDomainPrefTlv NASQmiCbkNasSystemSelPrefInd::SDPTlv

8.207 NASRoamPreferenceTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint16_t RoamPref

8.207.1 Field Documentation

8.207.1.1 uint16_t NASRoamPreferenceTlv::RoamPref

8.207.1.2 uint8_t NASRoamPreferenceTlv::TlvPresent

8.208 NASServDomainPrefTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t SrvDomainPref

8.208.1 Field Documentation

8.208.1.1 uint32_t NASServDomainPrefTlv::SrvDomainPref

8.208.1.2 uint8_t NASServDomainPrefTlv::TlvPresent

8.209 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

8.209.1 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

8.209.2 Field Documentation

8.209.2.1 uint8_t NASServingSystemInfo::csAttachState

8.209.2.2 uint8_t NASServingSystemInfo::hdrPersonality

8.209.2.3 uint8_t NASServingSystemInfo::psAttachState

8.209.2.4 uint8_t NASServingSystemInfo::radioInterfaceList[255]

8.209.2.5 uint8_t NASServingSystemInfo::radioInterfaceNo

8.209.2.6 uint8_t NASServingSystemInfo::registrationState

8.209.2.7 uint8_t NASServingSystemInfo::selectedNetwork

8.210 NASTimeInfoTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t time

8.210.1 Field Documentation

8.210.1.1 uint64_t NASTimeInfoTlv::time

8.210.1.2 uint8_t NASTimeInfoTlv::TlvPresent

8.211 newMTMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSMTMessageInfo MTMessageInfo

8.211.1 Detailed Description

Parameters

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

8.211.2 Field Documentation

8.211.2.1 sMSMTMessageInfo newMTMessageTlv::MTMessageInfo

8.211.2.2 uint8_t newMTMessageTlv::TlvPresent

8.212 pack_dms_ActivateAutomatic_t Struct Reference

Data Fields

- uint8_t actCode [81]

8.212.1 Detailed Description

This structure contains UIM activate automation information

Parameters

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

8.212.2 Field Documentation

8.212.2.1 uint8_t pack_dms_ActivateAutomatic_t::actCode[81]

8.213 pack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

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

8.213.1 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)
list_type	• list type requested
Tlvresult	• Pack Result

8.213.2 Field Documentation

8.213.2.1 uint8_t pack_dms_GetCustFeaturesV2_t::cust_id[64+1]

8.213.2.2 uint8_t pack_dms_GetCustFeaturesV2_t::list_type

8.213.2.3 uint16_t pack_dms_GetCustFeaturesV2_t::Tlvresult

8.214 pack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint8_t spc [6]

8.214.1 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)

8.214.2 Field Documentation

8.214.2.1 uint8_t pack_dms_ResetToFactoryDefaults_t::spc[6]

8.215 pack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint8_t activationState

8.215.1 Detailed Description

This structure is used to store Service Activation Status parameter .

Parameters

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

8.215.2 Field Documentation

8.215.2.1 uint8_t pack_dms_SetActivationStatusCallback_t::activationState

8.216 pack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t crashAction

8.216.1 Detailed Description

Modem action in case of a crash

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

8.216.2 Field Documentation

8.216.2.1 uint8_t pack_dms_SetCrashAction_t::crashAction

8.217 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

8.217.1 Field Documentation

8.217.1.1 uint8_t pack_dms_SetCustFeature_t::DHCPRelayEnabled

8.217.1.2 uint8_t pack_dms_SetCustFeature_t::DisableIMSI

8.217.1.3 uint32_t pack_dms_SetCustFeature_t::GpsEnable

8.217.1.4 uint8_t pack_dms_SetCustFeature_t::GPSLPM

8.217.1.5 uint8_t pack_dms_SetCustFeature_t::GPSSel

8.217.1.6 uint16_t pack_dms_SetCustFeature_t::IPFamSupport

8.217.1.7 uint8_t pack_dms_SetCustFeature_t::IsVoiceEnabled

8.217.1.8 uint8_t pack_dms_SetCustFeature_t::RMAutoConnect

8.217.1.9 uint8_t pack_dms_SetCustFeature_t::SMSSupport

8.218 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

8.218.1 Detailed Description

This structure contains customization settings set to modem pack

Parameters

cust_id	• Customization ID (Maximum 64 bytes)
value_length	• length of cust_value field
cust_value	• Customization Setting Value (Maximum 8 bytes)
Tlvresult	• Pack Result

8.218.2 Field Documentation

8.218.2.1 uint8_t pack_dms_SetCustFeaturesV2_t::cust_id[64+1]

8.218.2.2 uint8_t pack_dms_SetCustFeaturesV2_t::cust_value[8+1]

8.218.2.3 uint16_t pack_dms_SetCustFeaturesV2_t::Tlvresult

8.218.2.4 uint16_t pack_dms_SetCustFeaturesV2_t::value_length

8.219 pack_dms_SetEventReport_t Struct Reference

Data Fields

- uint8_t mode

8.219.1 Field Documentation

8.219.1.1 uint8_t pack_dms_SetEventReport_t::mode

8.220 pack_dms_SetPower_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t Tlvresult

8.220.1 Field Documentation

8.220.1.1 uint32_t pack_dms_SetPower_t::mode

8.220.1.2 uint16_t pack_dms_SetPower_t::Tlvresult

8.221 pack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint8_t USBComp
- uint16_t Tlvresult

8.221.1 Field Documentation

8.221.1.1 uint16_t pack_dms_SetUSBComp_t::Tlvresult

8.221.1.2 uint8_t pack_dms_SetUSBComp_t::USBComp

8.222 pack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint8_t resetInfoInd

8.222.1 Detailed Description

Parameters

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

8.222.2 Field Documentation

8.222.2.1 uint8_t pack_dms_SLQSDmsSwiIndicationRegister_t::resetInfoInd

8.223 pack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint8_t clear

8.223.1 Detailed Description

This structure contains crashInfoParams

Parameters

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

8.223.2 Field Documentation

8.223.2.1 uint8_t pack_dms_SLQSSwiGetCrashInfo_t::clear

8.224 pack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.224.1 Detailed Description

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.

8.224.2 Field Documentation

8.224.2.1 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSContent

8.224.2.2 uint8_t* pack_dms_SLQSSwiSetDyingGaspCfg_t::pDestSMSNum

8.225 pack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

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

8.225.1 Detailed Description

This structure contains SWI set host device info unpack information

Parameters

manString[IN]	• optional parameter, host device manufacture
modelString[IN]	• optional parameter, host device model
swVerString[IN]	• optional parameter, host device software version
plasmaIDString[IN]	• optional parameter, host device plasma ID

8.225.2 Field Documentation

8.225.2.1 char pack_dms_SLQSSwiSetHostDevInfo_t::manString[255]

8.225.2.2 char pack_dms_SLQSSwiSetHostDevInfo_t::modelString[255]

8.225.2.3 char pack_dms_SLQSSwiSetHostDevInfo_t::plasmaIDString[255]

8.225.2.4 char pack_dms_SLQSSwiSetHostDevInfo_t::swVerString[255]

8.226 pack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

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

8.226.1 Detailed Description

This structure contains SWI set host OS info pack information

Parameters

nameString[IN]	• optional parameter, host device manufacture
VersionString[IN]	• optional parameter, host device model

8.226.2 Field Documentation

8.226.2.1 char pack_dms_SLQSSwiSetOSInfo_t::nameString[255]

8.226.2.2 char pack_dms_SLQSSwiSetOSInfo_t::versionString[255]

8.227 pack_dms_UIMChangePIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t oldValue [255]
- uint8_t newValue [255]

8.227.1 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

8.227.2 Field Documentation

8.227.2.1 uint8_t pack_dms_UIMChangePIN_t::id

8.227.2.2 uint8_t pack_dms_UIMChangePIN_t::newValue[255]

8.227.2.3 uint8_t pack_dms_UIMChangePIN_t::oldValue[255]

8.228 pack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facility

8.228.1 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)

8.228.2 Field Documentation

8.228.2.1 uint8_t pack_dms_UIMGetControlKeyStatus_t::facility

8.229 pack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.229.1 Detailed Description

This structure contains UIM Get ICCID pack

Parameters

Tlvresult	• Pack result.

8.229.2 Field Documentation

8.229.2.1 uint16_t pack_dms_UIMGetICCID_t::Tlvresult

8.230 pack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

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

8.230.1 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
facilityCk[IN]	• Facility depersonalization control key, string in ASCII text (maximum 8 bytes)

8.230.2 Field Documentation

8.230.2.1 uint8_t pack_dms_UIMSetControlKeyProtection_t::facility

8.230.2.2 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityCk[8]

8.230.2.3 uint8_t pack_dms_UIMSetControlKeyProtection_t::facilityState

8.231 pack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

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

8.231.1 Detailed Description

Set activation status pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
in	*reqArg*	request parameter This structure contains PIN Protection Information
	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

8.231.2 Field Documentation

8.231.2.1 uint8_t pack_dms_UIMSetPINProtection_t::bEnable

8.231.2.2 uint8_t pack_dms_UIMSetPINProtection_t::id

8.231.2.3 uint8_t pack_dms_UIMSetPINProtection_t::value[255]

8.232 pack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t facility
- uint8_t facilityCk [8]

8.232.1 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)

8.232.2 Field Documentation

8.232.2.1 uint8_t pack_dms_UIMUnblockControlKey_t::facility

8.232.2.2 uint8_t pack_dms_UIMUnblockControlKey_t::facilityCk[8]

8.233 pack_dms_UIMUnblockPIN_t Struct Reference

Data Fields

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

8.233.1 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

8.233.2 Field Documentation

8.233.2.1 uint8_t pack_dms_UIMUnblockPIN_t::id

8.233.2.2 uint8_t pack_dms_UIMUnblockPIN_t::newPin[255]

8.233.2.3 uint8_t pack_dms_UIMUnblockPIN_t::pukValue[255]

8.234 pack_dms_UIMVerifyPIN_t Struct Reference

Data Fields

- uint8_t id
- uint8_t value [255]

8.234.1 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

8.234.2 Field Documentation

8.234.2.1 uint8_t pack_dms_UIMVerifyPIN_t::id

8.234.2.2 uint8_t pack_dms_UIMVerifyPIN_t::value[255]

8.235 pack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.235.1 Detailed Description

This structure contains the Get Image Preference information pack

Parameters

Tlvresult	• Pack result

8.235.2 Field Documentation

8.235.2.1 uint16_t pack_fms_GetImagesPreference_t::Tlvresult

8.236 pack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.236.1 Detailed Description

This structure contains the Get Stored Images pack

Parameters

Tlvresult	• Pack result

8.236.2 Field Documentation

8.236.2.1 uint16_t pack_fms_GetStoredImages_t::Tlvresult

8.237 pack_fms_SetImagesPreference_t Struct Reference

Data Fields

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

8.237.1 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

8.237.2 Field Documentation

8.237.2.1 uint32_t pack_fms_SetImagesPreference_t::bForceDownload

8.237.2.2 uint32_t pack_fms_SetImagesPreference_t::imageListSize

8.237.2.3 uint8_t pack_fms_SetImagesPreference_t::modemindex

8.237.2.4 FMSPrefImageList* pack_fms_SetImagesPreference_t::pImageList

8.237.2.5 uint16_t pack_fms_SetImagesPreference_t::Tlvresult

8.238 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

8.238.1 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.

8.238.2 Field Documentation

8.238.2.1 loc_BdsSVInfo* pack_loc_Delete_Assist_Data_t::pBdsSVInfo

8.238.2.2 loc_CellDb* pack_loc_Delete_Assist_Data_t::pCellDb

8.238.2.3 loc_ClkInfo* pack_loc_Delete_Assist_Data_t::pClkInfo

8.238.2.4 loc_GnssData* pack_loc_Delete_Assist_Data_t::pGnssData

8.238.2.5 loc_SVInfo* pack_loc_Delete_Assist_Data_t::pSVInfo

8.238.2.6 uint16_t pack_loc_Delete_Assist_Data_t::Tlvresult

8.239 pack_loc_EventRegister_t Struct Reference

Data Fields

- uint64_t eventRegister
- uint16_t Tlvresult

8.239.1 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.

8.239.2 Field Documentation

8.239.2.1 uint64_t pack_loc_EventRegister_t::eventRegister

8.239.2.2 uint16_t pack_loc_EventRegister_t::Tlvresult

8.240 pack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint32_t extPowerState
- uint16_t Tlvresult

8.240.1 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.

8.240.2 Field Documentation

8.240.2.1 uint32_t pack_loc_SetExtPowerState_t::extPowerState

8.240.2.2 uint16_t pack_loc_SetExtPowerState_t::Tlvresult

8.241 pack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint32_t mode
- uint16_t Tlvresult

8.241.1 Detailed Description

This structure contains Set Operation Mode pack

Parameters

mode	• 0 - Default Mode.
TlvResult	• Pack result.

8.241.2 Field Documentation

8.241.2.1 uint32_t pack_loc_SetOperationMode_t::mode

8.241.2.2 uint16_t pack_loc_SetOperationMode_t::TlvResult

8.242 pack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint32_t xid
- uint16_t TlvResult

8.242.1 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.

8.242.2 Field Documentation

8.242.2.1 uint16_t pack_loc_SLQSLOCGetBestAvailPos_t::TlvResult

8.242.2.2 uint32_t pack_loc_SLQSLOCGetBestAvailPos_t::xid

8.243 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

8.243.1 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

Parameters

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

Parameters

altitudeSrcInfo	• Optional parameter

- Pointer to struct altitudeSrcInfo. See altitudeSrcInfo for more information

Parameters

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.
rawHorUnc-Circular	• Optional parameter • Horizontal position uncertainty (circular) without any optimization. • Units - Meters
rawHor-Confidence	• 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.

8.243.2 Field Documentation

8.243.2.1 altSrcInfo_t pack_loc_SLQSLOCInjectPosition_t::altitudeSrcInfo

8.243.2.2 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtEllipsoid

8.243.2.3 float pack_loc_SLQSLOCInjectPosition_t::altitudeWrtMeanSeaLevel

8.243.2.4 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeSrcInfo

8.243.2.5 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtEllipsoid

8.243.2.6 int pack_loc_SLQSLOCInjectPosition_t::has_altitudeWrtMeanSeaLevel

8.243.2.7 int pack_loc_SLQSLOCInjectPosition_t::has_horConfidence

8.243.2.8 int pack_loc_SLQSLOCInjectPosition_t::has_horReliability

8.243.2.9 int pack_loc_SLQSLOCInjectPosition_t::has_horUncCircular

8.243.2.10 int pack_loc_SLQSLOCInjectPosition_t::has_latitude

8.243.2.11 int pack_loc_SLQSLOCInjectPosition_t::has_longitude

8.243.2.12 int pack_loc_SLQSLOCInjectPosition_t::has_positionSrc

8.243.2.13 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorConfidence

8.243.2.14 int pack_loc_SLQSLOCInjectPosition_t::has_rawHorUncCircular

8.243.2.15 int pack_loc_SLQSLOCInjectPosition_t::has_timestampAge

8.243.2.16 int pack_loc_SLQSLOCInjectPosition_t::has_timestampUtc

8.243.2.17 int pack_loc_SLQSLOCInjectPosition_t::has_vertConfidence

8.243.2.18 int pack_loc_SLQSLOCInjectPosition_t::has_vertRelicability

8.243.2.19 int pack_loc_SLQSLOCInjectPosition_t::has_vertUnc

8.243.2.20 uint8_t pack_loc_SLQSLOCInjectPosition_t::horConfidence

8.243.2.21 uint32_t pack_loc_SLQSLOCInjectPosition_t::horReliability

8.243.2.22 float pack_loc_SLQSLOCInjectPosition_t::horUncCircular

8.243.2.23 double pack_loc_SLQSLOCInjectPosition_t::latitude

8.243.2.24 double pack_loc_SLQSLOCInjectPosition_t::longitude

8.243.2.25 uint32_t pack_loc_SLQSLOCInjectPosition_t::positionSrc

8.243.2.26 uint8_t pack_loc_SLQSLOCInjectPosition_t::rawHorConfidence

8.243.2.27 float pack_loc_SLQSLOCInjectPosition_t::rawHorUncCircular

8.243.2.28 uint32_t pack_loc_SLQSLOCInjectPosition_t::timestampAge

8.243.2.29 uint64_t pack_loc_SLQSLOCInjectPosition_t::timestampUtc

8.243.2.30 uint8_t pack_loc_SLQSLOCInjectPosition_t::vertConfidence

8.243.2.31 uint32_t pack_loc_SLQSLOCInjectPosition_t::vertReliability

8.243.2.32 float pack_loc_SLQSLOCInjectPosition_t::vertUnc

8.244 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

8.244.1 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	• Optional parameter • Pointer to struct sensorData. See sensorData for more information
gyroData	• Optional parameter • Pointer to struct sensorData. See sensorData 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	• Optional parameter • Pointer to struct tempratureData. See tempratureData for more information

gyroTemp	• Optional parameter • Pointer to struct temprature Data. See temprature Data for more information

8.244.2 Field Documentation

8.244.2.1 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::acceleroData

8.244.2.2 uint32_t pack_loc_SLQSLOCInjectSensorData_t::acceleroTimeSrc

8.244.2.3 tempData_t pack_loc_SLQSLOCInjectSensorData_t::accelTemp

8.244.2.4 sensorData_t pack_loc_SLQSLOCInjectSensorData_t::gyroData

8.244.2.5 tempData_t pack_loc_SLQSLOCInjectSensorData_t::gyroTemp

8.244.2.6 uint32_t pack_loc_SLQSLOCInjectSensorData_t::gyroTimeSrc

8.244.2.7 int pack_loc_SLQSLOCInjectSensorData_t::has_acceleroTimeSrc

8.244.2.8 int pack_loc_SLQSLOCInjectSensorData_t::has_accelTemp

8.244.2.9 int pack_loc_SLQSLOCInjectSensorData_t::has_accleroData

8.244.2.10 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroData

8.244.2.11 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTemp

8.244.2.12 int pack_loc_SLQSLOCInjectSensorData_t::has_gyroTimeSrc

8.244.2.13 int pack_loc_SLQSLOCInjectSensorData_t::has_opaqueId

8.244.2.14 uint32_t pack_loc_SLQSLOCInjectSensorData_t::opaqueId

8.245 pack_loc_SLQSLOCInjectUTCTime_t Struct Reference

Data Fields

- uint64_t timeMsec
- uint32_t timeUncMsec

8.245.1 Detailed Description

Parameters

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

8.245.2 Field Documentation

8.245.2.1 uint64_t pack_loc_SLQSLOCInjectUTCTime_t::timeMsec

8.245.2.2 uint32_t pack_loc_SLQSLOCInjectUTCTime_t::timeUncMsec

8.246 pack_loc_SLQSLOCSetCradleMountConfig_t Struct Reference

Data Fields

- uint32_t state
- int has_confidence
- uint8_t confidence

8.246.1 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

8.246.2 Field Documentation

8.246.2.1 uint8_t pack_loc_SLQSLOCSetCradleMountConfig_t::confidence

8.246.2.2 int pack_loc_SLQSLOCSetCradleMountConfig_t::has_confidence

8.246.2.3 uint32_t pack_loc_SLQSLOCSetCradleMountConfig_t::state

8.247 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

8.247.1 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	• 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	• 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	• 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
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
LocApplication-Info	• LOC Application Parameters • See loc_LocApplicationInfo for more information
pConfigAltitude-Assumed	• 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

8.247.2 Field Documentation

8.247.2.1 loc_LocApplicationInfo* pack_loc_Start_t::pApplicationInfo

8.247.2.2 uint32_t* pack_loc_Start_t::pConfigAltitudeAssumed

8.247.2.3 uint32_t* pack_loc_Start_t::pHorizontalAccuracyLvl

8.247.2.4 uint32_t* pack_loc_Start_t::pIntermediateReportState

8.247.2.5 uint32_t* pack_loc_Start_t::pMinIntervalTime

8.247.2.6 uint32_t* pack_loc_Start_t::pRecurrenceType

8.247.2.7 uint8_t pack_loc_Start_t::SessionId

8.247.2.8 uint16_t pack_loc_Start_t::Tlvresult

8.248 pack_loc_Stop_t Struct Reference

Data Fields

- uint8_t SessionId
- uint16_t Tlvresult

8.248.1 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.

8.248.2 Field Documentation

8.248.2.1 uint8_t pack_loc_Stop_t::SessionId

8.248.2.2 uint16_t pack_loc_Stop_t::Tlvresult

8.249 pack_nas_InitiateDomainAttach_t Struct Reference

Data Fields

- uint32_t action

8.249.1 Detailed Description

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

Parameters

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

8.249.2 Field Documentation

8.249.2.1 uint32_t pack_nas_InitiateDomainAttach_t::action

8.250 pack_nas_SetACCOLC_t Struct Reference

Data Fields

- int8_t spc [6]
- uint8_t accolc

8.250.1 Detailed Description

Parameters

spc	servcie programming code
accolc	accolc

8.250.2 Field Documentation

8.250.2.1 uint8_t pack_nas_SetACCOLC_t::accolc

8.250.2.2 int8_t pack_nas_SetACCOLC_t::spc[6]

8.251 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

8.251.1 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

8.251.2 Field Documentation

8.251.2.1 uint32_t* pack_nas_SetCDMANetworkParameters_t::pApplication

8.251.2.2 uint32_t* pack_nas_SetCDMANetworkParameters_t::pBroadcast

8.251.2.3 uint8_t* pack_nas_SetCDMANetworkParameters_t::pCustomSCP

8.251.2.4 uint8_t* pack_nas_SetCDMANetworkParameters_t::pForceRev0

8.251.2.5 uint32_t* pack_nas_SetCDMANetworkParameters_t::pProtocol

8.251.2.6 uint32_t* pack_nas_SetCDMANetworkParameters_t::pRoaming

8.251.2.7 char* pack_nas_SetCDMANetworkParameters_t::pSPC

8.252 pack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint32_t TechnologyPref
- uint32_t Duration
- uint16_t Tlvresult

8.252.1 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

8.252.2 Field Documentation

8.252.2.1 uint32_t pack_nas_SetNetworkPreference_t::Duration

8.252.2.2 uint32_t pack_nas_SetNetworkPreference_t::TechnologyPref

8.252.2.3 uint16_t pack_nas_SetNetworkPreference_t::Tlvresult

8.253 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

8.253.1 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

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

8.253.2 Field Documentation

8.253.2.1 nas_ECIOThresh∗ pack_nas_SLQSConfigSigInfo_t::pECIOThresh

8.253.2.2 nas_HDRSINRThresh∗ pack_nas_SLQSConfigSigInfo_t::pHDRSINRThresh

8.253.2.3 nas_IOThresh∗ pack_nas_SLQSConfigSigInfo_t::pIOThresh

8.253.2.4 nas_LTESigRptCfg∗ pack_nas_SLQSConfigSigInfo_t::pLTESigRptCfg

8.253.2.5 nas_LTESNRThresh∗ pack_nas_SLQSConfigSigInfo_t::pLTESNRThresh

8.253.2.6 nas_RSRPThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRPThresh

8.253.2.7 nas_RSRQThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSRQThresh

8.253.2.8 nas_RSSIThresh∗ pack_nas_SLQSConfigSigInfo_t::pRSSIThresh

8.253.2.9 nas_TDSCDMASINRCONFThresh∗ pack_nas_SLQSConfigSigInfo_t::pTDSCDMASINRCONFThresh

8.254 pack_nas_SLQSGetPLMNName_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- uint8_t ∗ pMncPcsStatus

8.254.1 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.

8.254.2 Field Documentation

8.254.2.1 uint16_t pack_nas_SLQSGetPLMNName_t::mcc

8.254.2.2 uint16_t pack_nas_SLQSGetPLMNName_t::mnc

8.254.2.3 uint8_t* pack_nas_SLQSGetPLMNName_t::pMncPcsStatus

8.255 pack_nas_SLQSInitiateNetworkRegistration_t Struct Reference

Data Fields

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

8.255.1 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.

8.255.2 Field Documentation

8.255.2.1 uint32_t* pack_nas_SLQSInitiateNetworkRegistration_t::pChangeDuration

8.255.2.2 uint8_t* pack_nas_SLQSInitiateNetworkRegistration_t::pMncPcsDigitStatus

8.255.2.3 nas_MNRInfo* pack_nas_SLQSInitiateNetworkRegistration_t::pMNRInfo

8.255.2.4 uint32_t pack_nas_SLQSInitiateNetworkRegistration_t::regAction

8.256 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

8.256.1 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 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 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.

8.256.2 Field Documentation

8.256.2.1 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIODelta

8.256.2.2 nas_CDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMAECIOThresh

8.256.2.3 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIDelta

8.256.2.4 nas_CDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pCDMARSSIThresh

8.256.2.5 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIDelta

8.256.2.6 nas_GSMRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pGSMRSSIThresh

8.256.2.7 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIODelta

8.256.2.8 nas_HDRECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRECIOThresh

8.256.2.9 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIODelta

8.256.2.10 nas_HDRIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRIOThresh

8.256.2.11 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIDelta

8.256.2.12 nas_HDRRSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pHDRRSSIThresh

8.256.2.13 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRDelta

8.256.2.14 nas_HDRSINRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pHDRSINRThresh

8.256.2.15 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPDelta

8.256.2.16 nas_LTERSRPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRPThresh

8.256.2.17 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQDelta

8.256.2.18 nas_LTERSRQThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSRQThresh

8.256.2.19 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIDelta

8.256.2.20 nas_LTERSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pLTERSSIThresh

8.256.2.21 nas_LTESigRptConfig* pack_nas_SLQSNasConfigSigInfo2_t::pLTESigRptConfig

8.256.2.22 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRDelta

8.256.2.23 nas_LTESNRThreshold* pack_nas_SLQSNasConfigSigInfo2_t::pLTESNRThresh

8.256.2.24 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIODelta

8.256.2.25 nas_TDSCDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMAECIOThresh

8.256.2.26 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPDelta

8.256.2.27 nas_TDSCDMARSCPThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSCPThresh

8.256.2.28 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIDelta

8.256.2.29 nas_TDSCDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMARSSIThresh

8.256.2.30 float* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRDelta

8.256.2.31 nas_TDSCDMASINRThresh* pack_nas_SLQSNasConfigSigInfo2_t::pTDSCDMASINRThresh

8.256.2.32 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIODelta

8.256.2.33 nas_WCDMAECIOThresh* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMAECIOThresh

8.256.2.34 uint16_t* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIDelta

8.256.2.35 nas_WCDMARSSIThresh* pack_nas_SLQSNasConfigSigInfo2_t::pWCDMARSSIThresh

8.257 pack_nas_SLQSNasGet3GPP2Subscription_t Struct Reference

Data Fields

- uint8_t namID

8.257.1 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.

8.257.2 Field Documentation

8.257.2.1 uint8_t pack_nas_SLQSNasGet3GPP2Subscription_t::namID

8.258 pack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- uint8_t radio_if

8.258.1 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

8.258.2 Field Documentation

8.258.2.1 uint8_t pack_nas_SLQSNasGetTxRxInfo_t::radio_if

8.259 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

8.259.1 Detailed Description

Parameters

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

pSignalStrength-Ind	[Optional] • Signal Strength indication registration.The following callbacks would not be invoked if the indication is disabled. tFNSigInfo – 0x00 - Disable – 0x01 - Enable
pErrorRateInd	[Optional] • Error Rate indication registration.The following callbacks would not be invoked if the indication is disabled. tFNErrRate – 0x00 - Disable – 0x01 - Enable
pHDRNewUATI-AssInd	[Optional] • HDR New UATI Assigned indication registration.The following callbacks would not be invoked if the indication is disabled. tFNHDRUATIUpdate – 0x00 - Disable – 0x01 - Enable
pHDRSession-CloseInd	[Optional] • HDR Session Closed indication registration.The following callbacks would not be invoked if the indication is disabled. tFNHDRSessionClose – 0x00 - Disable – 0x01 - Enable
pManaged-RoamingInd	[Optional] • Managed Roaming indication registration.The following callbacks would not be invoked if the indication is disabled. tFNManagedRoaming – 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. tFNManagedRoaming – 0x00 - Disable (default value) – 0x01 - Enable

8.259.2 Field Documentation

8.259.2.1 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDDTMInd

8.259.2.2 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pDualStandByPrefInd

8.259.2.3 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pErrorRateInd

8.259.2.4 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRNewUATIAssInd

8.259.2.5 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pHDRSessionCloseInd

8.259.2.6 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pLTECphyCa

8.259.2.7 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pManagedRoamingInd

8.259.2.8 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkRejectInd

8.259.2.9 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pNetworkTimeInd

8.259.2.10 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pServingSystemInd

8.259.2.11 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSignalStrengthInd

8.259.2.12 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSubscriptionInfoInd

8.259.2.13 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSuppressSysInfoInd

8.259.2.14 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSysInfoInd

8.259.2.15 uint8_t* pack_nas_SLQSNasIndicationRegisterExt_t::pSystemSelectionInd

8.260 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

8.260.1 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

8.260.2 Field Documentation

8.260.2.1 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsDl

8.260.2.2 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::gsmUmtsUl

8.260.2.3 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmDl

8.260.2.4 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEmmUl

8.260.2.5 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmDl

8.260.2.6 uint8_t pack_nas_SLQSNasSwiIndicationRegister_t::lteEsmUl

8.260.2.7 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pRankIndicatorInd

8.260.2.8 uint8_t* pack_nas_SLQSNasSwiIndicationRegister_t::pTimer

8.261 pack_nas_SLQSNASSwiSetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN * pWcdmaUARFCN

- nas_lteEARFCN * pLteEARFCN
- nas_ltePCI * pLtePCI

8.261.1 Detailed Description

Structure for ChannelLock pack.

Parameters

pWcdmaUARFCN	[Optional] • See nas_wcdmaUARCFN for more information
pLteEARFCN	[Optional] • See nas_lteEARCFN for more information
pLtePCI	[Optional] • See nas_ltePCI for more information

8.261.2 Field Documentation

8.261.2.1 nas_lteEARFCN* pack_nas_SLQSNASSwiSetChannelLock_t::pLteEARFCN

8.261.2.2 nas_ltePCI* pack_nas_SLQSNASSwiSetChannelLock_t::pLtePCI

8.261.2.3 nas_wcdmaUARFCN* pack_nas_SLQSNASSwiSetChannelLock_t::pWcdmaUARFCN

8.262 pack_nas_SLQSSetBandPreference_t Struct Reference

Data Fields

- uint64_t bandPref

8.262.1 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

8.262.2 Field Documentation

8.262.2.1 uint64_t pack_nas_SLQSSetBandPreference_t::bandPref

8.263 pack_nas_SLQSSetSignalStrengthsCallback_t Struct Reference

Data Fields

- uint8_t bEnable
- nas_SLQSSignalStrengthsIndReq * pSigIndReq

8.263.1 Detailed Description

Parameters

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

8.263.2 Field Documentation

8.263.2.1 uint8_t pack_nas_SLQSSetSignalStrengthsCallback_t::bEnable

8.263.2.2 nas_SLQSSignalStrengthsIndReq* pack_nas_SLQSSetSignalStrengthsCallback_t::pSigIndReq

8.264 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

8.264.1 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 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

8.264.2 Field Documentation

8.264.2.1 struct nas_acqOrderPref* pack_nas_SLQSSetSysSelectionPref_t::pAcqOrderPref

8.264.2.2 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pBandPref

8.264.2.3 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pChgDuration

8.264.2.4 struct nas_CSGID* pack_nas_SLQSSetSysSelectionPref_t::pCSGID

8.264.2.5 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pEmerMode

8.264.2.6 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pGWAcqOrderPref

8.264.2.7 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pLTEBandPref

8.264.2.8 uint8_t* pack_nas_SLQSSetSysSelectionPref_t::pMNCIncPCSDigStat

8.264.2.9 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pModePref

8.264.2.10 struct nas_netSelectionPref* pack_nas_SLQSSetSysSelectionPref_t::pNetSelPref

8.264.2.11 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pPRLPref

8.264.2.12 unsigned char* pack_nas_SLQSSetSysSelectionPref_t::pRAT

8.264.2.13 uint16_t* pack_nas_SLQSSetSysSelectionPref_t::pRoamPref

8.264.2.14 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvDomainPref

8.264.2.15 uint32_t* pack_nas_SLQSSetSysSelectionPref_t::pSrvRegRestriction

8.264.2.16 uint64_t* pack_nas_SLQSSetSysSelectionPref_t::pTdscdmaBandPref

8.265 pack_nas_SLQSSwiPSDetach_t Struct Reference

Data Fields

- uint8_t * pDetachAction

8.265.1 Detailed Description

Structure for PS connection detach pack.

Parameters

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

8.265.2 Field Documentation

8.265.2.1 uint8_t* pack_nas_SLQSSwiPSDetach_t::pDetachAction

8.266 pack_pds_SetPDSDefaults_t Struct Reference

Data Fields

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

8.266.1 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)

8.266.2 Field Documentation

8.266.2.1 uint32_t pack_pds_SetPDSDefaults_t::accuracy

8.266.2.2 uint32_t pack_pds_SetPDSDefaults_t::interval

8.266.2.3 uint32_t pack_pds_SetPDSDefaults_t::operation

8.266.2.4 uint8_t pack_pds_SetPDSDefaults_t::timeout

8.267 pack_pds_SetPDSState_t Struct Reference

Data Fields

- uint32_t enable

8.267.1 Detailed Description

Structure contain PDS state parameters.

Parameters

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

8.267.2 Field Documentation

8.267.2.1 uint32_t pack_pds_SetPDSState_t::enable

8.268 pack_pds_SetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t bAuto

8.268.1 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

8.268.2 Field Documentation

8.268.2.1 uint32_t pack_pds_SetPortAutomaticTracking_t::bAuto

8.269 pack_qmi_t Struct Reference

Data Fields

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

8.269.1 Detailed Description

qmi request context

Parameters

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

8.269.2 Field Documentation

8.269.2.1 uint16_t pack_qmi_t::msgid

8.269.2.2 uint8_t pack_qmi_t::svc

8.269.2.3 int pack_qmi_t::timeout

8.269.2.4 uint16_t pack_qmi_t::xid

8.270 pack_qos_SLQSQosSwiReadApnExtraParams_t Struct Reference

Data Fields

- uint32_t apnId

8.270.1 Detailed Description

Structure that contains the APN ID to obtain extra APN parameters

Parameters

apnId[IN]	• APN id

8.270.2 Field Documentation

8.270.2.1 uint32_t pack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.271 pack_qos_SLQSQosSwiReadDataStats_t Struct Reference

Data Fields

- uint32_t apnId

8.271.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

apnId[IN]	• APN id

8.271.2 Field Documentation

8.271.2.1 uint32_t pack_qos_SLQSQosSwiReadDataStats_t::apnId

8.272 pack_qos_SLQSSetQosEventCallback_t Struct Reference

Data Fields

- uint8_t enable

8.272.1 Detailed Description

Structure that contains the APN ID to obtain data statistics

Parameters

enable[IN]	• 1 - Enable QoS event reporting • 0 - Disable QoS event reporting

8.272.2 Field Documentation

8.272.2.1 uint8_t pack_qos_SLQSSetQosEventCallback_t::enable

8.273 pack_rms_SetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask

8.273.1 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)

8.273.2 Field Documentation

8.273.2.1 uint32_t pack_rms_SetSMSWake_t::enabled

8.273.2.2 uint32_t pack_rms_SetSMSWake_t::wake_mask

8.274 pack_sar_SLQSSetRfSarState_t Struct Reference

Data Fields

- uint32_t RfSarState

8.274.1 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

8.274.2 Field Documentation

8.274.2.1 uint32_t pack_sar_SLQSSetRfSarState_t::RfSarState

8.275 pack_sms_SaveSMS_t Struct Reference

Data Fields

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

8.275.1 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

8.275.2 Field Documentation

8.275.2.1 uint32_t pack_sms_SaveSMS_t::messageFormat

8.275.2.2 uint32_t pack_sms_SaveSMS_t::messageSize

8.275.2.3 uint8_t* pack_sms_SaveSMS_t::pMessage

8.275.2.4 uint32_t pack_sms_SaveSMS_t::storageType

8.276 pack_sms_SendSMS_t Struct Reference

Data Fields

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

8.276.1 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

8.276.2 Field Documentation

8.276.2.1 uint32_t pack_sms_SendSMS_t::messageFormat

8.276.2.2 uint32_t pack_sms_SendSMS_t::messageSize

8.276.2.3 uint8_t* pack_sms_SendSMS_t::pLinktimer

8.276.2.4 uint8_t* pack_sms_SendSMS_t::pMessage

8.277 pack_sms_SetNewSMSCallback_t Struct Reference

Data Fields

- enum eqmiCbkSetStatus status

8.277.1 Detailed Description

Parameters

status	callback parameter

8.277.2 Field Documentation

8.277.2.1 enum eqmiCbkSetStatus pack_sms_SetNewSMSCallback_t::status

8.278 pack_sms_SetSMSCAddress_t Struct Reference

Data Fields

- uint8_t * pSMSCAddress
- uint8_t * pSMSCType

8.278.1 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).

8.278.2 Field Documentation

8.278.2.1 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCAddress

8.278.2.2 uint8_t* pack_sms_SetSMSCAddress_t::pSMSCType

8.279 pack_sms_SLQSDeleteSMS_t Struct Reference

Data Fields

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

8.279.1 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)

8.279.2 Field Documentation

8.279.2.1 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageIndex

8.279.2.2 uint8_t* pack_sms_SLQSDeleteSMS_t::pMessageMode

8.279.2.3 uint32_t* pack_sms_SLQSDeleteSMS_t::pMessageTag

8.279.2.4 uint32_t pack_sms_SLQSDeleteSMS_t::storageType

8.280 pack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex
- uint8_t * pMessageMode

8.280.1 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)

8.280.2 Field Documentation

8.280.2.1 uint32_t pack_sms_SLQSGetSMS_t::messageIndex

8.280.2.2 uint8_t* pack_sms_SLQSGetSMS_t::pMessageMode

8.280.2.3 uint32_t pack_sms_SLQSGetSMS_t::storageType

8.281 pack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode

8.281.1 Detailed Description

Parameters

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

8.281.2 Field Documentation

8.281.2.1 uint8_t pack_sms_SLQSGetSmsBroadcastConfig_t::mode

8.282 pack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t storageType
- uint32_t * pRequestedTag
- uint8_t * pMessageMode

8.282.1 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)

8.282.2 Field Documentation

8.282.2.1 uint8_t* pack_sms_SLQSGetSMSList_t::pMessageMode

8.282.2.2 uint32_t* pack_sms_SLQSGetSMSList_t::pRequestedTag

8.282.2.3 uint32_t pack_sms_SLQSGetSMSList_t::storageType

8.283 pack_sms_SLQSModifySMSStatus_t Struct Reference

Data Fields

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

8.283.1 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)

8.283.2 Field Documentation

8.283.2.1 uint32_t pack_sms_SLQSModifySMSStatus_t::messageIndex

8.283.2.2 uint32_t pack_sms_SLQSModifySMSStatus_t::messageTag

8.283.2.3 uint8_t* pack_sms_SLQSModifySMSStatus_t::pMessageMode

8.283.2.4 uint32_t pack_sms_SLQSModifySMSStatus_t::storageType

8.284 pack_sms_SLQSSendAsyncSMS_t Struct Reference

Data Fields

- sms_sendAsyncsmsParams * pSendSmsParams

8.284.1 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

8.284.2 Field Documentation

8.284.2.1 sms_sendAsyncsmsParams* pack_sms_SLQSSendAsyncSMS_t::pSendSmsParams

8.285 pack_sms_SLQSSetIndicationRegister_t Struct Reference

Data Fields

- sms_setIndicationReg * pSetIndicationRegReq

8.285.1 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

8.285.2 Field Documentation

8.285.2.1 sms_setIndicationReg* pack_sms_SLQSSetIndicationRegister_t::pSetIndicationRegReq

8.286 pack_sms_SLQSSetSmsBroadcastActivation_t Struct Reference

Data Fields

- uint8_t mode
- uint8_t broadcastActivate

8.286.1 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

8.286.2 Field Documentation

8.286.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::broadcastActivate

8.286.2.2 uint8_t pack_sms_SLQSSetSmsBroadcastActivation_t::mode

8.287 pack_sms_SLQSSetSmsBroadcastConfig_t Struct Reference

Data Fields

- uint8_t mode
- sms_qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig
- sms_qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig

8.287.1 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).

8.287.2 Field Documentation

8.287.2.1 uint8_t pack_sms_SLQSSetSmsBroadcastConfig_t::mode

8.287.2.2 sms_qaQmi3GPPBroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pBroadcastConfig

8.287.2.3 sms_qaQmi3GPP2BroadcastCfgInfo* pack_sms_SLQSSetSmsBroadcastConfig_t::pCDMABroadcastConfig

8.288 pack_sms_SLQSSetSmsStorage_t Struct Reference

Data Fields

- uint8_t smsStorage

8.288.1 Detailed Description

Structure contain Parameter to set the SMS Storage on the device

Parameters

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

8.288.2 Field Documentation

8.288.2.1 uint8_t pack_sms_SLQSSetSmsStorage_t::smsStorage

8.289 pack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeReq * pMaxStorageSizeReq

8.289.1 Detailed Description

Parameters

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

8.289.2 Field Documentation

8.289.2.1 sms_maxStorageSizeReq* pack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeReq

8.290 pack_sms_SLQSSmsSetRoutes_t Struct Reference

Data Fields

- sms_setRoutesReq * pSetRoutesReq

8.290.1 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

8.290.2 Field Documentation

8.290.2.1 sms_setRoutesReq* pack_sms_SLQSSmsSetRoutes_t::pSetRoutesReq

8.291 pack_swiavms_SLQSAvMSSendSelection_t Struct Reference

Data Fields

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

8.291.1 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.

pClientPerformOperationFlag[IN/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.

8.291.2 Field Documentation

8.291.2.1 uint8_t* pack_swiavms_SLQSAvMSSendSelection_t::pClientPerformOperationFlag

8.291.2.2 uint32_t* pack_swiavms_SLQSAvMSSendSelection_t::pDeferTime

8.291.2.3 uint8_t* pack_swiavms_SLQSAvMSSendSelection_t::pPackageID

8.291.2.4 uint16_t* pack_swiavms_SLQSAvMSSendSelection_t::pRejectReason

8.291.2.5 uint8_t pack_swiavms_SLQSAvMSSendSelection_t::selection

8.292 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

8.292.1 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

8.292.2 Field Documentation

8.292.2.1 uint8_t pack_swiavms_SLQSAvMSSetSettings_t::AutoConnect

8.292.2.2 uint8_t pack_swiavms_SLQSAvMSSetSettings_t::AutoReboot

8.292.2.3 uint8_t pack_swiavms_SLQSAvMSSetSettings_t::FwAutodownload

8.292.2.4 uint8_t pack_swiavms_SLQSAvMSSetSettings_t::FwAutoUpdate

8.292.2.5 PackSwiAvmsSetSettingsAPNInfo* pack_swiavms_SLQSAvMSSetSettings_t::pAPNInfo

8.292.2.6 PackSwiAvmsSetSettingsConnectionRetryTimers* pack_swiavms_SLQSAvMSSetSettings_t::pConnectionRetryTimers

8.292.2.7 uint8_t* pack_swiavms_SLQSAvMSSetSettings_t::pFwAutoSDM

8.292.2.8 uint8_t* pack_swiavms_SLQSAvMSSetSettings_t::pNotifStore

8.292.2.9 PackSwiAvmsSetSettingsPeriodInfo* pack_swiavms_SLQSAvMSSetSettings_t::pPeriodInfo

8.292.2.10 uint32_t* pack_swiavms_SLQSAvMSSetSettings_t::pPollingTimer

8.293 pack_swiavms_SLQSAvMSStartSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.293.1 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.

8.293.2 Field Documentation

8.293.2.1 uint8_t pack_swiavms_SLQSAvMSStartSession_t::sessionType

8.294 pack_swiavms_SLQSAvMSStopSession_t Struct Reference

Data Fields

- uint8_t sessionType

8.294.1 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

8.294.2 Field Documentation

8.294.2.1 uint8_t pack_swiavms_SLQSAvMSStopSession_t::sessionType

8.295 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

8.295.1 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 • I - 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 • I - 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

8.295.2 Field Documentation

8.295.2.1 uint32_t pack_swiloc_SwiLocSetAutoStart_t::fix_rate

8.295.2.2 uint8_t pack_swiloc_SwiLocSetAutoStart_t::fix_type

8.295.2.3 uint8_t pack_swiloc_SwiLocSetAutoStart_t::function

8.295.2.4 uint32_t pack_swiloc_SwiLocSetAutoStart_t::max_dist

8.295.2.5 uint8_t pack_swiloc_SwiLocSetAutoStart_t::max_time

8.295.2.6 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_rate

8.295.2.7 int pack_swiloc_SwiLocSetAutoStart_t::set_fix_type

8.295.2.8 int pack_swiloc_SwiLocSetAutoStart_t::set_function

8.295.2.9 int pack_swiloc_SwiLocSetAutoStart_t::set_max_dist

8.295.2.10 int pack_swiloc_SwiLocSetAutoStart_t::set_max_time

8.296 pack_swioma_SLQSOMADMCancelSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.296.1 Detailed Description

Structure that contains the session type for OMA cancel session command

Parameters

sessionType[IN]	• Session type – 0x01 - FOTA, to check availability of FW Update – 0xFF - Cancel any active OMADM session

8.296.2 Field Documentation

8.296.2.1 uint32_t pack_swioma_SLQSOMADMCancelSession_t::sessionType

8.297 pack_swioma_SLQSOMADMGetSessionInfo_t Struct Reference

Data Fields

- uint32_t SessionType

8.297.1 Detailed Description

Structure that contains the session type for OMA get session info command

Parameters

SessionType[IN]	• Session type – 0x01 - FOTA – 0xFF - Any active OMADM session. If no active sessions are available, then previous OMADM session info is returned

8.297.2 Field Documentation

8.297.2.1 uint32_t pack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.298 pack_swioma_SLQSOMADMSendSelection_t Struct Reference

Data Fields

- uint32_t selection
- uint32_t * pDeferTime
- uint32_t * pRejectReason

8.298.1 Detailed Description

Structure containing the OMA DM selection

Parameters

selection[IN]	• OMA-DM NIA Selection – 0x01 - Accept – 0x02 - Reject – 0x03 - Defer
pDeferTime[IN]	• 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.
pRejectReason[IN]	• 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.

8.298.2 Field Documentation

8.298.2.1 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pDeferTime

8.298.2.2 uint32_t* pack_swioma_SLQSOMADMSendSelection_t::pRejectReason

8.298.2.3 uint32_t pack_swioma_SLQSOMADMSendSelection_t::selection

8.299 pack_swioma_SLQSOMADMSetSettings_t Struct Reference

Data Fields

- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t * pAutosdm
- uint8_t * pFwAutoCheck

8.299.1 Detailed Description

Structure containing the OMA DM settings to be set on the device This maps to structure SLQSOMADMSettingsReqParams3

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[IN]	• Optional 1 byte parameter indicating OMA Automatic UI Alert Response – 0x00 - Disabled – 0x01 - Enabled Accept – 0x02 - Enabled Reject
pFwAutoCheck[-IN]	• Optional 1 byte parameter indicating OMA Automatic Check for Firmware Update on Power-Up Response – 0x00 - Disabled – 0x01 - Enabled

8.299.2 Field Documentation

8.299.2.1 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAdownload

8.299.2.2 uint8_t pack_swioma_SLQSOMADMSetSettings_t::FOTAUpdate

8.299.2.3 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pAutosdm

8.299.2.4 uint8_t* pack_swioma_SLQSOMADMSetSettings_t::pFwAutoCheck

8.300 pack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t sessionType

8.300.1 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

8.300.2 Field Documentation

8.300.2.1 uint32_t pack_swioma_SLQSOMADMStartSession_t::sessionType

8.301 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

8.301.1 Detailed Description

This structure contains information of the request parameters associated with a Change PIN API.

Parameters

sessionInfo	• See UIMSessionInformation for more information.
changePIN	• See 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.

8.301.2 Field Documentation

8.301.2.1 uim_changeUIMPIN pack_uim_ChangePin_t::changePIN

8.301.2.2 uim_encryptedPIN1 pack_uim_ChangePin_t::EncryptedPIN1

8.301.2.3 uint32_t* pack_uim_ChangePin_t::pIndicationToken

8.301.2.4 uint8_t* pack_uim_ChangePin_t::pKeyReferenceID

8.301.2.5 uim_sessionInformation pack_uim_ChangePin_t::sessionInfo

8.301.2.6 uint16_t pack_uim_ChangePin_t::Tlvresult

8.302 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

8.302.1 Detailed Description

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

Parameters

sessionInfo	• See UIMSessionInformation for more information.
fileIndex	• See fileInfo for more information.
readTransparent	• See 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.

8.302.2 Field Documentation

8.302.2.1 uim_fileInfo pack_uim_ReadTransparent_t::fileIndex

8.302.2.2 uint8_t* pack_uim_ReadTransparent_t::pEncryptData

8.302.2.3 uint32_t* pack_uim_ReadTransparent_t::pIndicationToken

8.302.2.4 uim_readTransparentInfo pack_uim_ReadTransparent_t::readTransparent

8.302.2.5 uim_sessionInformation pack_uim_ReadTransparent_t::sessionInfo

8.302.2.6 uint16_t pack_uim_ReadTransparent_t::TlvResult

8.303 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

8.303.1 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.

8.303.2 Field Documentation

8.303.2.1 uim_encryptedPIN1 pack_uim_SetPinProtection_t::EncryptedPIN1

8.303.2.2 uint32_t* pack_uim_SetPinProtection_t::pIndicationToken

8.303.2.3 uim_setPINProtection pack_uim_SetPinProtection_t::pinProtection

8.303.2.4 uint8_t* pack_uim_SetPinProtection_t::pKeyReferenceID

8.303.2.5 uim_sessionInformation pack_uim_SetPinProtection_t::sessionInfo

8.303.2.6 uint16_t pack_uim_SetPinProtection_t::TlvResult

8.304 pack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_authenticationData authData
- uint32_t * pIndicationToken

8.304.1 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.

8.304.2 Field Documentation

8.304.2.1 uim_authenticationData pack_uim_SLQSUIMAuthenticate_t::authData

8.304.2.2 uint32_t* pack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.304.2.3 uim_sessionInformation pack_uim_SLQSUIMAuthenticate_t::sessionInfo

8.305 pack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_depersonalizationInformation depersonilisationInfo

8.305.1 Detailed Description

This structure contains information of the request parameters associated with a Depersonalization.

Parameters

depersonilisation-Info	• See uim_depersonalizationInformation for more information.

8.305.2 Field Documentation

8.305.2.1 uim_depersonalizationInformation pack_uim_SLQSUIMDepersonalization_t::depersonilisationInfo

8.306 pack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask

8.306.1 Detailed Description

Parameters

eventMask	- bit 1 - card status • bit 4 - physical slot status

8.306.2 Field Documentation

8.306.2.1 uint32_t pack_uim_SLQSUIMEventRegister_t::eventMask

8.307 pack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint32_t * pConfigurationMask

8.307.1 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.

8.307.2 Field Documentation

8.307.2.1 uint32_t* pack_uim_SLQSUIMGetConfiguration_t::pConfigurationMask

8.308 pack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_fileInfo fileIndex
- uint32_t * pIndicationToken

8.308.1 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.

8.308.2 Field Documentation

8.308.2.1 uim_fileInfo pack_uim_SLQSUIMGetFileAttributes_t::fileIndex

8.308.2.2 uint32_t* pack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.308.2.3 uim_sessionInformation pack_uim_SLQSUIMGetFileAttributes_t::sessionInfo

8.309 pack_uim_SLQSUIMPowerDown_t Struct Reference

Data Fields

- uint8_t slot

8.309.1 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

8.309.2 Field Documentation

8.309.2.1 uint8_t pack_uim_SLQSUIMPowerDown_t::slot

8.310 pack_uim_SLQSUIMPowerUp_t Struct Reference

Data Fields

- uint8_t slot
- uint8_t * pIgnoreHotSwapSwitch

8.310.1 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

8.310.2 Field Documentation

8.310.2.1 uint8_t* pack_uim_SLQSUIMPowerUp_t::pIgnoreHotSwapSwitch

8.310.2.2 uint8_t pack_uim_SLQSUIMPowerUp_t::slot

8.311 pack_uim_SLQSUIMRefreshComplete_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t refreshComplete

8.311.1 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

8.311.2 Field Documentation

8.311.2.1 uint8_t pack_uim_SLQSUIMRefreshComplete_t::refreshComplete

8.311.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshComplete_t::sessionInfo

8.312 pack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo

8.312.1 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.

8.312.2 Field Documentation

8.312.2.1 uim_sessionInformation pack_uim_SLQSUIMRefreshGetLastEvent_t::sessionInfo

8.313 pack_uim_SLQSUIMRefreshOK_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uint8_t OKtoRefresh

8.313.1 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

8.313.2 Field Documentation

8.313.2.1 uint8_t pack_uim_SLQSUIMRefreshOK_t::OKtoRefresh

8.313.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshOK_t::sessionInfo

8.314 pack_uim_SLQSUIMRefreshRegister_t Struct Reference

Data Fields

- uim_sessionInformation sessionInfo
- uim_registerRefresh regRefresh

8.314.1 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

8.314.2 Field Documentation

8.314.2.1 uim_registerRefresh pack_uim_SLQSUIMRefreshRegister_t::regRefresh

8.314.2.2 uim_sessionInformation pack_uim_SLQSUIMRefreshRegister_t::sessionInfo

8.315 pack_uim_SLQSUIMSwitchSlot_t Struct Reference

Data Fields

- uint8_t bLogicalSlot
- uint32_t ulPhysicalSlot

8.315.1 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

8.315.2 Field Documentation

8.315.2.1 uint8_t pack_uim_SLQSUIMSwitchSlot_t::bLogicalSlot

8.315.2.2 uint32_t pack_uim_SLQSUIMSwitchSlot_t::ulPhysicalSlot

8.316 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

8.316.1 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.

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

8.316.2 Field Documentation

8.316.2.1 uim_encryptedPIN1 pack_uim_UnblockPin_t::EncryptedPIN1

8.316.2.2 uint32_t* pack_uim_UnblockPin_t::pIndicationToken

8.316.2.3 uim_unblockUIMPIN pack_uim_UnblockPin_t::pinProtection

8.316.2.4 uint8_t* pack_uim_UnblockPin_t::pKeyReferenceID

8.316.2.5 uim_sessionInformation pack_uim_UnblockPin_t::sessionInfo

8.316.2.6 uint16_t pack_uim_UnblockPin_t::Tlvresult

8.317 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

8.317.1 Detailed Description

This structure contains information of the request parameters associated with a verify PIN API.

Parameters

sessionInfo	• See UIMSessionInformation for more information.
verifyPIN	• See verifyUIMPIN for more information.

pEncryptedPIN1(optional)	• See 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.

8.317.2 Field Documentation

8.317.2.1 uim_encryptedPIN1∗ pack_uim_VerifyPin_t::pEncryptedPIN1

8.317.2.2 uint32_t∗ pack_uim_VerifyPin_t::pIndicationToken

8.317.2.3 uint8_t∗ pack_uim_VerifyPin_t::pKeyReferenceID

8.317.2.4 uim_sessionInformation pack_uim_VerifyPin_t::sessionInfo

8.317.2.5 uint16_t pack_uim_VerifyPin_t::Tlvresult

8.317.2.6 uim_verifyUIMPIN pack_uim_VerifyPin_t::verifyPIN

8.318 pack_voice_AnswerUSSD_t Struct Reference

Data Fields

- uint8_t ∗ pInfo

8.318.1 Detailed Description

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.318.2 Field Documentation

8.318.2.1 uint8_t* pack_voice_AnswerUSSD_t::pInfo

8.319 pack_voice_OriginateUSSD_t Struct Reference

Data Fields

- uint8_t * pInfo

8.319.1 Detailed Description

Parameters

pInfo	• USS information • See voice_USSInfo for more details

8.319.2 Field Documentation

8.319.2.1 uint8_t* pack_voice_OriginateUSSD_t::pInfo

8.320 pack_voice_SLQSOriginateUSSD_t Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.320.1 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

8.320.2 Field Documentation

8.320.2.1 uint8_t pack_voice_SLQSOriginateUSSD_t::ussData[182]

8.320.2.2 uint8_t pack_voice_SLQSOriginateUSSD_t::ussDCS

8.320.2.3 uint8_t pack_voice_SLQSOriginateUSSD_t::ussLen

8.321 pack_voice_SLQSVoiceALSSelectLine_t Struct Reference

Data Fields

- uint8_t lineValue

8.321.1 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

8.321.2 Field Documentation

8.321.2.1 uint8_t pack_voice_SLQSVoiceALSSelectLine_t::lineValue

8.322 pack_voice_SLQSVoiceALSSetLineSwitching_t Struct Reference

Data Fields

- uint8_t switchOption

8.322.1 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

8.322.2 Field Documentation

8.322.2.1 uint8_t pack_voice_SLQSVoiceALSSetLineSwitching_t::switchOption

8.323 pack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.323.1 Detailed Description

Contains the parameters passed for pack voice Answer Call.

Parameters

pCallId	• Unique call identifier for the call that must be answered.

8.323.2 Field Documentation

8.323.2.1 uint8_t* pack_voice_SLQSVoiceAnswerCall_t::pCallId

8.324 pack_voice_SLQSVoiceBindSubscription_t Struct Reference

Data Fields

- uint8_t subsType

8.324.1 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

8.324.2 Field Documentation

8.324.2.1 uint8_t pack_voice_SLQSVoiceBindSubscription_t::subsType

8.325 pack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- voice_burstDTMFInfo BurstDTMFInfo
- voice_DTMFLengths * pBurstDTMFLengths

8.325.1 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

8.325.2 Field Documentation

8.325.2.1 voice_burstDTMFInfo pack_voice_SLQSVoiceBurstDTMF_t::BurstDTMFInfo

8.325.2.2 voice_DTMFLengths* pack_voice_SLQSVoiceBurstDTMF_t::pBurstDTMFLengths

8.326 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

8.326.1 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

8.326.2 Field Documentation

8.326.2.1 uint8_t pack_voice_SLQSVoiceDialCall_t::callNumber[81]

8.326.2.2 voice_calledPartySubAdd* pack_voice_SLQSVoiceDialCall_t::pCallPartySubAdd

8.326.2.3 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCallType

8.326.2.4 uint8_t* pack_voice_SLQSVoiceDialCall_t::pCLIRType

8.326.2.5 voice_CUGInfo* pack_voice_SLQSVoiceDialCall_t::pCUGInfo

8.326.2.6 uint8_t* pack_voice_SLQSVoiceDialCall_t::pEmergencyCategory

8.326.2.7 uint8_t* pack_voice_SLQSVoiceDialCall_t::pSvcType

8.326.2.8 voice_UUSInfo* pack_voice_SLQSVoiceDialCall_t::pUUSInFo

8.327 pack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.327.1 Detailed Description

Parameters

pCallId	• Unique call identifier for the call that must be ended

8.327.2 Field Documentation

8.327.2.1 uint8_t* pack_voice_SLQSVoiceEndCall_t::pCallId

8.328 pack_voice_SLQSVoiceGetCallBarring_t Struct Reference

Data Fields

- uint8_t reason
- uint8_t * pSvcClass

8.328.1 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

8.328.2 Field Documentation

8.328.2.1 uint8_t* pack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.328.2.2 uint8_t pack_voice_SLQSVoiceGetCallBarring_t::reason

8.329 pack_voice_SLQSVoiceGetCallForwardingStatus_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t * pSvcClass

8.329.1 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.

8.329.2 Field Documentation

8.329.2.1 uint8_t* pack_voice_SLQSVoiceGetCallForwardingStatus_t::pSvcClass

8.329.2.2 uint8_t pack_voice_SLQSVoiceGetCallForwardingStatus_t::Reason

8.330 pack_voice_SLQSVoiceGetCallInfo_t Struct Reference

Data Fields

- uint8_t callID

8.330.1 Detailed Description

This structure contains information of the request parameters associated with a call.

Parameters

callID	• Call identifier for the call queried for information.

8.330.2 Field Documentation

8.330.2.1 uint8_t pack_voice_SLQSVoiceGetCallInfo_t::callID

8.331 pack_voice_SLQSVoiceGetCallWaiting_t Struct Reference

Data Fields

- uint8_t * pSvcClass

8.331.1 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

8.331.2 Field Documentation

8.331.2.1 uint8_t* pack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.332 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

8.332.1 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.

8.332.2 Field Documentation

8.332.2.1 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAirTimer

8.332.2.2 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAMRStatus

8.332.2.3 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pAutoAnswer

8.332.2.4 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pNamID

8.332.2.5 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoicePrivacy

8.332.2.6 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pPrefVoiceSO

8.332.2.7 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pRoamTimer

8.332.2.8 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pTTYMode

8.332.2.9 uint8_t* pack_voice_SLQSVoiceGetConfig_t::pVoiceDomainPref

8.333 pack_voice_SLQSVoiceIndicationRegister_t Struct Reference

Data Fields

- uint8_t * pRegDTMFEvents
- uint8_t * pRegVoicePrivacyEvents
- uint8_t * pSuppsNotifEvents

8.333.1 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.

8.333.2 Field Documentation

8.333.2.1 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegDTMFEvents

8.333.2.2 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pRegVoicePrivacyEvents

8.333.2.3 uint8_t* pack_voice_SLQSVoiceIndicationRegister_t::pSuppsNotifEvents

8.334 pack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint8_t SUPSType
- uint8_t * pCallID

8.334.1 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

8.334.2 Field Documentation

8.334.2.1 uint8_t* pack_voice_SLQSVoiceManageCalls_t::pCallID

8.334.2.2 uint8_t pack_voice_SLQSVoiceManageCalls_t::SUPSType

8.335 pack_voice_SLQSVoiceOrigUSSDNoWait_t Struct Reference

Data Fields

- struct voice_USSInfo USSInformation

8.335.1 Detailed Description

This structure contains Orig USSD No Wait Information Parameters.

Parameters

USSInformation	• See voice_USSInfo for more information.

8.335.2 Field Documentation

8.335.2.1 struct voice_USSInfo pack_voice_SLQSVoiceOrigUSSDNoWait_t::USSInformation

8.336 pack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t * pFlashPayLd
- uint8_t * pFlashType

8.336.1 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

8.336.2 Field Documentation

8.336.2.1 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pCallID

8.336.2.2 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashPayLd

8.336.2.3 uint8_t* pack_voice_SLQSVoiceSendFlash_t::pFlashType

8.337 pack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference

Data Fields

- uint8_t Reason
- uint8_t oldPasswd [4]
- uint8_t newPasswd [4]
- uint8_t newPasswdAgain [4]

8.337.1 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.

8.337.2 Field Documentation

8.337.2.1 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswd[4]

8.337.2.2 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::newPasswdAgain[4]

8.337.2.3 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::oldPasswd[4]

8.337.2.4 uint8_t pack_voice_SLQSVoiceSetCallBarringPassword_t::Reason

8.338 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

8.338.1 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 vocie_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.

8.338.2 Field Documentation

8.338.2.1 voice_airTimer* pack_voice_SLQSVoiceSetConfig_t::pAirTimerConfig

8.338.2.2 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pAutoAnswer

8.338.2.3 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceDomain

8.338.2.4 voice_prefVoiceSO* pack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSO

8.338.2.5 voice_roamTimer* pack_voice_SLQSVoiceSetConfig_t::pRoamTimerConfig

8.338.2.6 uint8_t* pack_voice_SLQSVoiceSetConfig_t::pTTYMode

8.339 pack_voice_SLQSVoiceSetPreferredPrivacy_t Struct Reference

Data Fields

- uint8_t privacyPref

8.339.1 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

8.339.2 Field Documentation

8.339.2.1 uint8_t pack_voice_SLQSVoiceSetPreferredPrivacy_t::privacyPref

8.340 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

8.340.1 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 serviceClassInformation 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

8.340.2 Field Documentation

8.340.2.1 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallBarringPasswd

8.340.2.2 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pCallForwardingNumber

8.340.2.3 voice_callFwdTypeAndPlan* pack_voice_SLQSVoiceSetSUPSService_t::pCallFwdTypeAndPlan

8.340.2.4 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pServiceClass

8.340.2.5 uint8_t* pack_voice_SLQSVoiceSetSUPSService_t::pTimerVal

8.340.2.6 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::reason

8.340.2.7 uint8_t pack_voice_SLQSVoiceSetSUPSService_t::voiceSvc

8.341 pack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t DTMFdigit

8.341.1 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.

8.341.2 Field Documentation

8.341.2.1 uint8_t pack_voice_SLQSVoiceStartContDTMF_t::DTMFdigit

8.341.2.2 uint8_t* pack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.342 pack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID

8.342.1 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

8.342.2 Field Documentation

8.342.2.1 uint8_t pack_voice_SLQSVoiceStopContDTMF_t::callID

8.343 pack_wds_DHCPv4ClientLeaseChange_t Struct Reference

Data Fields

- uint8_t * pEnableNotification

8.343.1 Detailed Description

WDS SWI DHCPv4 Client Lease Change Structure

Parameters

pEnable-Notification	• Enable Notification or not

8.343.2 Field Documentation

8.343.2.1 uint8_t* pack_wds_DHCPv4ClientLeaseChange_t::pEnableNotification

8.344 pack_wds_GetDefaultProfile_t Struct Reference

Data Fields

- uint32_t profiletype

8.344.1 Detailed Description

Parameters

profiletype	profile type

8.344.2 Field Documentation

8.344.2.1 uint32_t pack_wds_GetDefaultProfile_t::profiletype

8.345 pack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family

8.345.1 Detailed Description

Parameters

type	profile type • 0 - 3GPP • 1 - 3GPP2
type	profile family • 0 - Embedded • 1 - Tethered

8.345.2 Field Documentation

8.345.2.1 uint8_t pack_wds_GetDefaultProfileNum_t::family

8.345.2.2 uint8_t pack_wds_GetDefaultProfileNum_t::type

8.346 pack_wds_GetDormancyState_t Struct Reference

8.347 pack_wds_GetLastMobileIPError_t Struct Reference

8.348 pack_wds_GetMobileIP_t Struct Reference

8.349 pack_wds_GetMobileIPProfile_t Struct Reference

Data Fields

- uint8_t index

8.349.1 Detailed Description

Parameters

index	mobile ip profile identifier

8.349.2 Field Documentation

8.349.2.1 uint8_t pack_wds_GetMobileIPProfile_t::index

8.350 pack_wds_GetPacketStatistics_t Struct Reference

Data Fields

- uint32_t * pStatMask

8.350.1 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

8.350.2 Field Documentation

8.350.2.1 uint32_t* pack_wds_GetPacketStatistics_t::pStatMask

8.351 pack_wds_GetPacketStatus_t Struct Reference

Data Fields

- uint32_t statmask

8.351.1 Detailed Description

Parameters

statmask	packet statistics mask

8.351.2 Field Documentation

8.351.2.1 uint32_t pack_wds_GetPacketStatus_t::statmask

8.352 pack_wds_GetSessionDuration_t Struct Reference

8.353 pack_wds_RMSetTransferStatistics_t Struct Reference

Data Fields

- rmTrasnferStaticsReq RmTrasnferStaticsReq

8.353.1 Detailed Description

Parameters

rmTrasnfer-StaticsReq	RM Transfer Statistics Indicator

8.353.2 Field Documentation

8.353.2.1 rmTrasnferStaticsReq pack_wds_RMSetTransferStatistics_t::RmTrasnferStaticsReq

8.354 pack_wds_SetAutoconnect_t Struct Reference

Data Fields

- uint8_t acsetting
- uint8_t acroamsetting

8.354.1 Detailed Description

auto connect data session parameters.

Parameters

setting	• NDIS autoconnect setting – 0 - Disabled – 1 - Enabled

8.354.2 Field Documentation

8.354.2.1 uint8_t pack_wds_SetAutoconnect_t::acroamsetting

8.354.2.2 uint8_t pack_wds_SetAutoconnect_t::acsetting

8.355 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

8.355.1 Detailed Description

Parameters

profileType	profile type
pdpType	Packet Data Protocol (PDP) type
ipAddress	ip address
primaryDNS	primary dns
secondaryDNS	secondry dns
authentication	authentication type
name	name of the profile
apnname	apn name
username	username of the profile
password	password of profile

8.355.2 Field Documentation

8.355.2.1 uint32_t pack_wds_SetDefaultProfile_t::authentication

8.355.2.2 uint32_t pack_wds_SetDefaultProfile_t::ipAddress

8.355.2.3 uint8_t* pack_wds_SetDefaultProfile_t::pApnname

8.355.2.4 uint32_t pack_wds_SetDefaultProfile_t::pdpType

8.355.2.5 uint8_t* pack_wds_SetDefaultProfile_t::pName

8.355.2.6 uint8_t* pack_wds_SetDefaultProfile_t::pPassword

8.355.2.7 uint32_t pack_wds_SetDefaultProfile_t::primaryDNS

8.355.2.8 uint32_t pack_wds_SetDefaultProfile_t::profileType

8.355.2.9 uint8_t* pack_wds_SetDefaultProfile_t::pUsername

8.355.2.10 uint32_t pack_wds_SetDefaultProfile_t::secondaryDNS

8.356 pack_wds_SetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t type
- uint8_t family
- uint8_t index

8.356.1 Field Documentation

8.356.1.1 uint8_t pack_wds_SetDefaultProfileNum_t::family

8.356.1.2 uint8_t pack_wds_SetDefaultProfileNum_t::index

8.356.1.3 uint8_t pack_wds_SetDefaultProfileNum_t::type

8.357 pack_wds_SetMobileIP_t Struct Reference

Data Fields

- uint32_t mode

8.357.1 Detailed Description

Parameters

mode	• Mobile IP setting – 0 - Mobile IP off (simple IP only) – 1 - Mobile IP preferred – 2 - Mobile IP only

8.357.2 Field Documentation

8.357.2.1 uint32_t pack_wds_SetMobileIP_t::mode

8.358 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

8.358.1 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

8.358.2 Field Documentation

8.358.2.1 uint8_t* pack_wds_SetMobileIPParameters_t::pHA2002bis

8.358.2.2 uint8_t* pack_wds_SetMobileIPParameters_t::pHAAuthenticator

8.358.2.3 uint32_t* pack_wds_SetMobileIPParameters_t::pMode

8.358.2.4 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegPeriod

8.358.2.5 uint8_t* pack_wds_SetMobileIPParameters_t::pReRegTraffic

8.358.2.6 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryInterval

8.358.2.7 uint8_t* pack_wds_SetMobileIPParameters_t::pRetryLimit

8.358.2.8 char* pack_wds_SetMobileIPParameters_t::pSPC

8.359 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

8.359.1 Detailed Description

Parameters

spc	service programming code string
index	Index of the profile to modify
pEnabled	Enable profile 0-disable nonzero enable
pAddress	Home IPv4 address
pPrimaryHA	Primary home agent IPv4 address
pSecondaryHA	secondary home agent IPv4 address
pRevTunneling	Enable reverse tunneling 0-disable nonzero enable
pNAI	Network access identifier string
pHASPI	Home agent security parameter index
pAAASPI	AAA server security parameter index
pMNHA	MN-HA key string
pMNAAA	MN-AAA key string

8.359.2 Field Documentation

8.359.2.1 uint8_t pack_wds_SetMobileIPProfile_t::index

8.359.2.2 uint32_t* pack_wds_SetMobileIPProfile_t::pAAASPI

8.359.2.3 uint32_t* pack_wds_SetMobileIPProfile_t::pAddress

8.359.2.4 uint8_t* pack_wds_SetMobileIPProfile_t::pEnabled

8.359.2.5 uint32_t* pack_wds_SetMobileIPProfile_t::pHASPI

8.359.2.6 int8_t* pack_wds_SetMobileIPProfile_t::pMNAAA

8.359.2.7 int8_t* pack_wds_SetMobileIPProfile_t::pMNHA

8.359.2.8 int8_t* pack_wds_SetMobileIPProfile_t::pNAI

8.359.2.9 uint32_t* pack_wds_SetMobileIPProfile_t::pPrimaryHA

8.359.2.10 uint8_t* pack_wds_SetMobileIPProfile_t::pRevTunneling

8.359.2.11 uint32_t* pack_wds_SetMobileIPProfile_t::pSecondaryHA

8.359.2.12 int8_t pack_wds_SetMobileIPProfile_t::spc[10]

8.360 pack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo * pCurProfile

8.360.1 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

8.360.2 Field Documentation

8.360.2.1 wds_profileInfo* pack_wds_SLQSCreateProfile_t::pCurProfile

8.360.2.2 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileId

8.360.2.3 uint8_t* pack_wds_SLQSCreateProfile_t::pProfileType

8.361 pack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.361.1 Detailed Description

Parameters

profileType	profile type
profileIndex	profile index

8.361.2 Field Documentation

8.361.2.1 uint8_t pack_wds_SLQSDeleteProfile_t::profileIndex

8.361.2.2 uint8_t pack_wds_SLQSDeleteProfile_t::profileType

8.362 pack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

8.363 pack_wds_SLQSGetDataBearerTechnology_t Struct Reference

8.364 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

8.364.1 Detailed Description

Parameters

Mask	mast bits corresponding to the information requested to 1
pReportConnStatus	Connect Status Indicator
pTransferStatInd	Transfer Statistics Indicator
pReportDormStatus	Dormancy Status Indicator
pReportDataBearerTech	Current Data Bearer Technology Indicator
pReportChannelRate	Channel Rate Indicator

8.364.2 Field Documentation

8.364.2.1 uint32_t pack_wds_SLQSGetDUNCallInfo_t::Mask

8.364.2.2 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportChannelRate

8.364.2.3 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportConnStatus

8.364.2.4 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDataBearerTech

8.364.2.5 uint8_t* pack_wds_SLQSGetDUNCallInfo_t::pReportDormStatus

8.364.2.6 transferStatInd* pack_wds_SLQSGetDUNCallInfo_t::pTransferStatInd

8.365 pack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- uint8_t ProfileId
- uint8_t ProfileType

8.365.1 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

8.365.2 Field Documentation

8.365.2.1 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileId

8.365.2.2 uint8_t pack_wds_SLQSGetProfileSettings_t::ProfileType

8.366 pack_wds_SLQSGetRuntimeSettings_t Struct Reference

Data Fields

- uint32_t * pReqSettings

8.366.1 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

8.366.2 Field Documentation

8.366.2.1 uint32_t* pack_wds_SLQSGetRuntimeSettings_t::pReqSettings

8.367 pack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint8_t * pProfileId
- uint8_t * pProfileType
- wds_profileInfo curProfile

8.367.1 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

8.367.2 Field Documentation

8.367.2.1 wds_profileInfo pack_wds_SLQSModifyProfile_t::curProfile

8.367.2.2 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileId

8.367.2.3 uint8_t* pack_wds_SLQSModifyProfile_t::pProfileType

8.368 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

8.368.1 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

8.368.2 Field Documentation

8.368.2.1 uint16_t pack_wds_SLQSSet3GPPConfigItem_t::LTEAttachProfileListLen

8.368.2.2 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::p3gppRelease

8.368.2.3 uint8_t* pack_wds_SLQSSet3GPPConfigItem_t::pDefaultPDNEnabled

8.368.2.4 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfile

8.368.2.5 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pLTEAttachProfileList

8.368.2.6 uint16_t* pack_wds_SLQSSet3GPPConfigItem_t::pProfileList

8.369 pack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint8_t IPFamilyPreference

8.369.1 Detailed Description

Parameters

IPFamily-Preference	IP Family preference • PACK_WDS_IPV4 IP Version 4 • PACK_WDS_IPV6 IP Version 6

8.369.2 Field Documentation

8.369.2.1 uint8_t pack_wds_SLQSSetIPFamilyPreference_t::IPFamilyPreference

8.370 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

8.370.1 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

8.370.2 Field Documentation

8.370.2.1 uint8_t pack_wds_SLQSSetWdsEventCallback_t::currentDataBearer

8.370.2.2 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearer

8.370.2.3 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataBearerTechExt

8.370.2.4 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dataSystemStatus

8.370.2.5 uint8_t pack_wds_SLQSSetWdsEventCallback_t::dormancyStatus

8.370.2.6 uint8_t pack_wds_SLQSSetWdsEventCallback_t::interval

8.370.2.7 uint8_t pack_wds_SLQSSetWdsEventCallback_t::mobileIP

8.370.2.8 uint8_t pack_wds_SLQSSetWdsEventCallback_t::transferStats

8.371 pack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4ProfileId * pProfileId

8.371.1 Detailed Description

Parameters

pProfileId	pointer to Profile Id structure

8.371.2 Field Documentation

8.371.2.1 wdsDhcpv4ProfileId* pack_wds_SLQSSGetDHCPv4ClientConfig_t::pProfileId

8.372 pack_wds_SLQSSSetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wds_DHCPv4ProfileId * pProfileId
- wds_DHCPv4HWConfig * pHwConfig
- wds_DHCPv4OptionList * pRequestOptionList

8.372.1 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

8.372.2 Field Documentation

8.372.2.1 wds_DHCPv4HWConfig* pack_wds_SLQSSSetDHCPv4ClientConfig_t::pHwConfig

8.372.2.2 wds_DHCPv4ProfileId* pack_wds_SLQSSSetDHCPv4ClientConfig_t::pProfileId

8.372.2.3 wds_DHCPv4OptionList* pack_wds_SLQSSSetDHCPv4ClientConfig_t::pRequestOptionList

8.373 pack_wds_SLQSSSetLoopback_t Struct Reference

Data Fields

- uint8_t loopbackMode
- uint8_t loopbackMultiplier

8.373.1 Detailed Description

Parameters

loopbackMode	• Loopback Mode. – 0 - Disable – 1 - Enable
loopback-Multiplier	• Loopback multiplier. Number of downlink bytes to send for each uplink byte.

8.373.2 Field Documentation

8.373.2.1 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMode

8.373.2.2 uint8_t pack_wds_SLQSSSetLoopback_t::loopbackMultiplier

8.374 pack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint8_t * pTech
- uint32_t * pprofileid3gpp
- uint32_t * pprofileid3gpp2
- uint32_t * pAuth
- char * pUser
- char * pPass

8.374.1 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 3GPPs 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

8.374.2 Field Documentation

8.374.2.1 uint32_t* pack_wds_SLQSStartDataSession_t::pAuth

8.374.2.2 char* pack_wds_SLQSStartDataSession_t::pPass

8.374.2.3 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp

8.374.2.4 uint32_t* pack_wds_SLQSStartDataSession_t::pprofileid3gpp2

8.374.2.5 uint8_t* pack_wds_SLQSStartDataSession_t::pTech

8.374.2.6 char* pack_wds_SLQSStartDataSession_t::pUser

8.375 pack_wds_SLQSStopDataSession_t Struct Reference

Data Fields

- uint32_t * psid

8.375.1 Detailed Description

Parameters

sid	session id

8.375.2 Field Documentation

8.375.2.1 uint32_t* pack_wds_SLQSStopDataSession_t::psid

8.376 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

8.376.1 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.

8.376.2 Field Documentation

8.376.2.1 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrChannelRateInd

8.376.2.2 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrDataBearerTechInd

8.376.2.3 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pCurrPrefDataSysInd

8.376.2.4 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataBearerTechInd

8.376.2.5 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataCallStatusChangeInd

8.376.2.6 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDataSystemStatusChangeInd

8.376.2.7 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pDormancyStatusInd

8.376.2.8 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pEVDOPageMonPerChangeInd

8.376.2.9 uint8_t* pack_wds_SLQSWdsSetEventReport_t::pMIPStatusInd

8.376.2.10 wds_TrStatInd* pack_wds_SLQSWdsSetEventReport_t::pTransferStatInd

8.377 pack_wds_SLQSWdsSwiPDPRuntimeSettings_t Struct Reference

Data Fields

- uint8_t contextId
- uint8_t contextType

8.377.1 Detailed Description

Parameters

contextId	Context Identifier
contextType	Context Type 0-3GPP 1-3GPP2

8.377.2 Field Documentation

8.377.2.1 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.377.2.2 uint8_t pack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextType

8.378 PackCreateProfileOut Struct Reference

Data Fields

- uint8_t ProfileType
- uint8_t ProfileIndex
- uint16_t ExtErrorCode

8.378.1 Field Documentation

8.378.1.1 uint16_t PackCreateProfileOut::ExtErrorCode

8.378.1.2 uint8_t PackCreateProfileOut::ProfileIndex

8.378.1.3 uint8_t PackCreateProfileOut::ProfileType

8.379 packgetDyingGaspCfg Struct Reference

Data Fields

- uint8_t * pDestSMSNum
- uint8_t * pDestSMSContent

8.379.1 Detailed Description

Parameters

pDestSMSNum[-IN]	• SMS Destination Number as string of 8 bit ASCII Characters Max 20 chars. • Optional parameter.

pDestSMS-Content[IN]	• SMS COntent as a string of 8 bit ASCII text characters Max 160 chars. • Optional parameter.

8.379.2 Field Documentation

8.379.2.1 uint8_t* packgetDyingGaspCfg::pDestSMSContent

8.379.2.2 uint8_t* packgetDyingGaspCfg::pDestSMSNum

8.380 packgetDyingGaspStatistics Struct Reference

Data Fields

- uint32_t * pTimeStamp
- uint8_t * pSMSAttemptedFlag

8.380.1 Detailed Description

Parameters

TimeStamp[OU-T]	• Time Stamp.
SMSAttempted-Flag[OUT]	• SMS Attempted Flag.

8.380.2 Field Documentation

8.380.2.1 uint8_t* packgetDyingGaspStatistics::pSMSAttemptedFlag

8.380.2.2 uint32_t* packgetDyingGaspStatistics::pTimeStamp

8.381 qmiSmsMessageList Struct Reference

Data Fields

- uint32_t messageIndex
- uint32_t messageTag

8.381.1 Detailed Description

Parameters

messageIndex	• Message index of each matched message
messageTag	• Messagetag

8.381.2 Field Documentation

8.381.2.1 uint32_t qmiSmsMessageList::messageIndex

8.381.2.2 uint32_t qmiSmsMessageList::messageTag

8.382 qmiWDSDataBearerTechnology Struct Reference

Data Fields

- uint8_t currentNetwork
- uint32_t ratMask
- uint32_t soMask

8.382.1 Detailed Description

Parameters

currentNetwork	current selected network
Radio	Access Technology (RAT) mask
soMask	Service Option (SO) mask

8.382.2 Field Documentation

8.382.2.1 uint8_t qmiWDSDataBearerTechnology::currentNetwork

8.382.2.2 uint32_t qmiWDSDataBearerTechnology::ratMask

8.382.2.3 uint32_t qmiWDSDataBearerTechnology::soMask

8.383 RFBandInfoElements Struct Reference

Data Fields

- uint8_t radioInterface
- uint16_t activeBandClass
- uint16_t activeChannel

8.383.1 Detailed Description

Parameters

radioInterface	radio interface technology
activeBandClass	active band class
activeChannel	active channel

8.383.2 Field Documentation

8.383.2.1 uint16_t RFBandInfoElements::activeBandClass

8.383.2.2 uint16_t RFBandInfoElements::activeChannel

8.383.2.3 uint8_t RFBandInfoElements::radioInterface

8.384 rmTrasnferStaticsReq Struct Reference

Data Fields

- uint8_t bResetStatistics
- uint32_t ulMask

8.384.1 Detailed Description

Parameters

bResetStatistics	Clear RM statistics
ulMask	Requested statistic bit mask

8.384.2 Field Documentation

8.384.2.1 uint8_t rmTrasnferStaticsReq::bResetStatistics

8.384.2.2 uint32_t rmTrasnferStaticsReq::ulMask

8.385 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]

8.385.1 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

8.385.2 Field Documentation

8.385.2.1 uint8_t sensorData_t::flags

8.385.2.2 uint8_t sensorData_t::sensorDataLen

8.385.2.3 uint32_t sensorData_t::timeOfFirstSample

8.385.2.4 uint16_t sensorData_t::timeOffset[64]

8.385.2.5 uint32_t sensorData_t::xAxis[64]

8.385.2.6 uint32_t sensorData_t::yAxis[64]

8.385.2.7 uint32_t sensorData_t::zAxis[64]

8.386 slot_t Struct Reference

Data Fields

- uint32_t uPhyCardStatus
- uint32_t uPhySlotStatus
- uint8_t bLogicalSlot
- uint8_t bICCIDLength
- uint8_t bICCID [255]

8.386.1 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
bICCID[MAX_ICCID_LENGTH]	• Contains the ICCID of the card in the physical slot.

8.386.2 Field Documentation

8.386.2.1 uint8_t slot_t::bICCID[255]

8.386.2.2 uint8_t slot_t::bICCIDLength

8.386.2.3 uint8_t slot_t::bLogicalSlot

8.386.2.4 uint32_t slot_t::uPhyCardStatus

8.386.2.5 uint32_t slot_t::uPhySlotStatus

8.387 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]

8.387.1 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.

8.387.2 Field Documentation

8.387.2.1 appStats slotInf::AppStatus[10]

8.387.2.2 uint8_t slotInf::cardState

8.387.2.3 uint8_t slotInf::errorState

8.387.2.4 uint8_t slotInf::numApp

8.387.2.5 uint8_t slotInf::upinRetries

8.387.2.6 uint8_t slotInf::upinState

8.387.2.7 uint8_t slotInf::upukRetries

8.388 slots_t Struct Reference

Data Fields

- slot_t uimSlotStatus [255]

8.388.1 Field Documentation

8.388.1.1 slot_t slots_t::uimSlotStatus[255]

8.389 sms_BroadcastConfig Struct Reference

Data Fields

- uint16_t fromServiceId
- uint16_t toServiceId
- uint8_t selected

8.389.1 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

8.389.2 Field Documentation

8.389.2.1 uint16_t sms_BroadcastConfig::fromServiceId

8.389.2.2 uint8_t sms_BroadcastConfig::selected

8.389.2.3 uint16_t sms_BroadcastConfig::toServiceId

8.390 sms_CDMABroadcastConfig Struct Reference

Data Fields

- uint16_t serviceCategory
- uint16_t language
- uint8_t selected

8.390.1 Detailed Description

This structure contains CDMABroadcastConfig parameters

Parameters

serviceCategory	• Service category
language	• Language
selected	• Specified service_category and language – 0x00 – Not selected – 0x01 – Selected

8.390.2 Field Documentation

8.390.2.1 uint16_t sms_CDMABroadcastConfig::language

8.390.2.2 uint8_t sms_CDMABroadcastConfig::selected

8.390.2.3 uint16_t sms_CDMABroadcastConfig::serviceCategory

8.391 sms_getIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.391.1 Detailed Description

This structure contains Get Indication Register Response parameters

Parameters

pRegTransLayerInfoEvt	- • 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.
pRegTransNWRegInfoEvt	- • 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.
pRegCallStatInfoEvt	- • 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.

8.391.2 Field Documentation

8.391.2.1 uint8_t* sms_getIndicationReg::pRegCallStatInfoEvt

8.391.2.2 uint8_t* sms_getIndicationReg::pRegTransLayerInfoEvt

8.391.2.3 uint8_t* sms_getIndicationReg::pRegTransNWRegInfoEvt

8.392 sms_getMsgWaitingInfo Struct Reference

Data Fields

- uint8_t numInstances
- sms_messageWaitingInfoContent msgWaitInfo [0xFF]

8.392.1 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.

8.392.2 Field Documentation

8.392.2.1 sms_messageWaitingInfoContent sms_getMsgWaitingInfo::msgWaitInfo[0xFF]

8.392.2.2 uint8_t sms_getMsgWaitingInfo::numInstances

8.393 sms_getTransLayerInfo Struct Reference

Data Fields

- uint8_t * pRegInd
- sms_transLayerInfo * pTransLayerInfo

8.393.1 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.

8.393.2 Field Documentation

8.393.2.1 uint8_t* sms_getTransLayerInfo::pRegInd

8.393.2.2 sms_transLayerInfo* sms_getTransLayerInfo::pTransLayerInfo

8.394 sms_getTransNWRegInfo Struct Reference

Data Fields

- uint8_t * pRegStatus

8.394.1 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.

8.394.2 Field Documentation

8.394.2.1 uint8_t* sms_getTransNWRegInfo::pRegStatus

8.395 sms_maxStorageSizeReq Struct Reference

Data Fields

- uint8_t storageType
- uint8_t * pMessageMode

8.395.1 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

8.395.2 Field Documentation

8.395.2.1 uint8_t* sms_maxStorageSizeReq::pMessageMode

8.395.2.2 uint8_t sms_maxStorageSizeReq::storageType

8.396 sms_maxStorageSizeResp Struct Reference

Data Fields

- uint32_t maxStorageSize
- uint32_t freeSlots

8.396.1 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.

8.396.2 Field Documentation

8.396.2.1 uint32_t sms_maxStorageSizeResp::freeSlots

8.396.2.2 uint32_t sms_maxStorageSizeResp::maxStorageSize

8.397 sms_messageWaitingInfoContent Struct Reference

Data Fields

- uint8_t msgType
- uint8_t activeInd
- uint8_t msgCount

8.397.1 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

8.397.2 Field Documentation

8.397.2.1 uint8_t sms_messageWaitingInfoContent::activeInd

8.397.2.2 uint8_t sms_messageWaitingInfoContent::msgCount

8.397.2.3 uint8_t sms_messageWaitingInfoContent::msgType

8.398 sms_msgProtocolResp Struct Reference

Data Fields

- uint8_t msgProtocol

8.398.1 Detailed Description

This structure contains get message protocol response parameters

Parameters

msgProtocol	- • Message Protocol • Values: – 0x00 - MESSAGE_PROTOCOL_CDMA – 0x01 - MESSAGE_PROTOCOL_WCDMA

8.398.2 Field Documentation

8.398.2.1 uint8_t sms_msgProtocolResp::msgProtocol

8.399 sms_qaQmi3GPP2BroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_CDMABroadcastConfig CDMABroadcastConfig [0x05]

8.399.1 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

8.399.2 Field Documentation

8.399.2.1 uint8_t sms_qaQmi3GPP2BroadcastCfgInfo::activated_ind

8.399.2.2 sms_CDMABroadcastConfig sms_qaQmi3GPP2BroadcastCfgInfo::CDMABroadcastConfig[0x05]

8.399.2.3 uint16_t sms_qaQmi3GPP2BroadcastCfgInfo::num_instances

8.400 sms_qaQmi3GPPBroadcastCfgInfo Struct Reference

Data Fields

- uint8_t activated_ind
- uint16_t num_instances
- sms_BroadcastConfig broadcastConfig [0x05]

8.400.1 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

8.400.2 Field Documentation

8.400.2.1 uint8_t sms_qaQmi3GPPBroadcastCfgInfo::activated_ind

8.400.2.2 sms_BroadcastConfig sms_qaQmi3GPPBroadcastCfgInfo::broadcastConfig[0x05]

8.400.2.3 uint16_t sms_qaQmi3GPPBroadcastCfgInfo::num_instances

8.401 sms_routeEntry Struct Reference

Data Fields

- uint8_t messageType
- uint8_t messageClass
- uint8_t routeStorage
- uint8_t receiptAction

8.401.1 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)

8.401.2 Field Documentation

8.401.2.1 uint8_t sms_routeEntry::messageClass

8.401.2.2 uint8_t sms_routeEntry::messageType

8.401.2.3 uint8_t sms_routeEntry::receiptAction

8.401.2.4 uint8_t sms_routeEntry::routeStorage

8.402 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

8.402.1 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.

8.402.2 Field Documentation

8.402.2.1 uint32_t sms_sendAsyncsmsParams::messageFormat

8.402.2.2 uint32_t sms_sendAsyncsmsParams::messageSize

8.402.2.3 uint8_t* sms_sendAsyncsmsParams::pFollowOnDC

8.402.2.4 uint8_t* sms_sendAsyncsmsParams::pForceOnDC

8.402.2.5 uint8_t* sms_sendAsyncsmsParams::pLinktimer

8.402.2.6 uint8_t* sms_sendAsyncsmsParams::pMessage

8.402.2.7 uint8_t* sms_sendAsyncsmsParams::pRetryMessage

8.402.2.8 uint32_t* sms_sendAsyncsmsParams::pRetryMessageId

8.402.2.9 uint8_t* sms_sendAsyncsmsParams::pServiceOption

8.402.2.10 uint8_t* sms_sendAsyncsmsParams::pSmsOnIms

8.402.2.11 uint32_t* sms_sendAsyncsmsParams::pUserData

8.403 sms_setIndicationReg Struct Reference

Data Fields

- uint8_t * pRegTransLayerInfoEvt
- uint8_t * pRegTransNWRegInfoEvt
- uint8_t * pRegCallStatInfoEvt

8.403.1 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

8.403.2 Field Documentation

8.403.2.1 uint8_t* sms_setIndicationReg::pRegCallStatInfoEvt

8.403.2.2 uint8_t* sms_setIndicationReg::pRegTransLayerInfoEvt

8.403.2.3 uint8_t* sms_setIndicationReg::pRegTransNWRegInfoEvt

8.404 sms_setRoutesReq Struct Reference

Data Fields

- uint16_t numOfRoutes
- sms_routeEntry routeList [0x0A]
- uint8_t * pTransferStatusReport

8.404.1 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)

8.404.2 Field Documentation

8.404.2.1 uint16_t sms_setRoutesReq::numOfRoutes

8.404.2.2 uint8_t* sms_setRoutesReq::pTransferStatusReport

8.404.2.3 sms_routeEntry sms_setRoutesReq::routeList[0x0A]

8.405 sms_transLayerInfo Struct Reference

Data Fields

- uint8_t TransType
- uint8_t TransCap

8.405.1 Detailed Description

This structure contains Transport Layer Information

Parameters

TransType	• Transport Type – 0x00 - IMS
TransCap	• Transport Capability • Values: – 0x00 - CDMA – 0x01 - GW

8.405.2 Field Documentation

8.405.2.1 uint8_t sms_transLayerInfo::TransCap

8.405.2.2 uint8_t sms_transLayerInfo::TransType

8.406 sMSCAddress Struct Reference

Data Fields

- uint8_t length
- uint8_t data [256]

8.406.1 Detailed Description

Parameters

length	• Number of sets of following element
data	• SMSC address

8.406.2 Field Documentation

8.406.2.1 uint8_t sMSCAddress::data[256]

8.406.2.2 uint8_t sMSCAddress::length

8.407 sMSCAddressTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSCAddressInfo SMSCInfo

8.407.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
SMSCInfo	• SMSC Address • See sMSCAddressInfo for more information

8.407.2 Field Documentation

8.407.2.1 sMSCAddressInfo sMSCAddressTlv::SMSCInfo

8.407.2.2 uint8_t sMSCAddressTlv::TlvPresent

8.408 sMSEtwsMessage Struct Reference

Data Fields

- uint8_t notificationType
- uint16_t length
- uint8_t data [1254]

8.408.1 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

8.408.2 Field Documentation

8.408.2.1 uint8_t sMSEtwsMessage::data[1254]

8.408.2.2 uint16_t sMSEtwsMessage::length

8.408.2.3 uint8_t sMSEtwsMessage::notificationType

8.409 sMSEtwsMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSEtwsMessageInfo EtwsMessageInfo

8.409.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
EtwsMessage-Info	• ETWS Message • See sMSEtwsMessageInfo for more information

8.409.2 Field Documentation

8.409.2.1 sMSEtwsMessageInfo sMSEtwsMessageTlv::EtwsMessageInfo

8.409.2.2 uint8_t sMSEtwsMessageTlv::TlvPresent

8.410 sMSEtwsPlmn Struct Reference

Data Fields

- uint16_t mobileCountryCode
- uint16_t mobileNetworkCode

8.410.1 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

8.410.2 Field Documentation

8.410.2.1 uint16_t sMSEtwsPlmn::mobileCountryCode

8.410.2.2 uint16_t sMSEtwsPlmn::mobileNetworkCode

8.411 sMSMessageMode Struct Reference

Data Fields

- uint8_t messageMode

8.411.1 Detailed Description

Parameters

messageMode	Message Mode

8.411.2 Field Documentation

8.411.2.1 uint8_t sMSMessageMode::messageMode

8.412 sMSMTMessage Struct Reference

Data Fields

- uint32_t storageType
- uint32_t messageIndex

8.412.1 Detailed Description

Parameters

storageType	memory storage 0x00-UIM 0x01-NV
messageIndex	MT Message index

8.412.2 Field Documentation

8.412.2.1 uint32_t sMSMTMessage::messageIndex

8.412.2.2 uint32_t sMSMTMessage::storageType

8.413 sMSOnIMS Struct Reference

Data Fields

- uint8_t smsOnIMS

8.413.1 Detailed Description

Parameters

smsOnIMS	SMS on IMS

8.413.2 Field Documentation

8.413.2.1 uint8_t sMSOnIMS::smsOnIMS

8.414 sMSOnIMSTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSOnIMSInfo IMSInfo

8.414.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
IMSInfo	• SMS on IMS • See sMSOnIMSInfo for more information

8.414.2 Field Documentation

8.414.2.1 sMSOnIMSInfo sMSOnIMSTlv::IMSInfo

8.414.2.2 uint8_t sMSOnIMSTlv::TlvPresent

8.415 sMSTransferRouteMTMessage Struct Reference

Data Fields

- uint8_t ackIndicator
- uint32_t transactionID
- uint8_t format
- uint16_t length
- uint8_t data [256]

8.415.1 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

8.415.2 Field Documentation

8.415.2.1 uint8_t sMSTransferRouteMTMessage::ackIndicator

8.415.2.2 uint8_t sMSTransferRouteMTMessage::data[256]

8.415.2.3 uint8_t sMSTransferRouteMTMessage::format

8.415.2.4 uint16_t sMSTransferRouteMTMessage::length

8.415.2.5 uint32_t sMSTransferRouteMTMessage::transactionID

8.416 tdscdmaSigInfoExt Struct Reference

Data Fields

- float rssi
- float rscp
- float ecio
- float sinr

8.416.1 Detailed Description

Parameters

rssi	RSSI in dBm.
rsrq	RSRQ value in dB
rsrp	Current RSRP in dBm as measured by L1.
snr	SNR level as a scaled integer in units of 0.1 dB.

8.416.2 Field Documentation

8.416.2.1 float tdscdmaSigInfoExt::ecio

8.416.2.2 float tdscdmaSigInfoExt::rscp

8.416.2.3 float tdscdmaSigInfoExt::rssi

8.416.2.4 float tdscdmaSigInfoExt::sinr

8.417 tempData_t Struct Reference

Data Fields

- uint32_t timeSource
- uint32_t timeOfFirstSample
- uint8_t temperatureDataLen
- uint16_t timeOffset [64]
- uint32_t temperature [64]

8.417.1 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

8.417.2 Field Documentation

8.417.2.1 uint32_t tempData_t::temperature[64]

8.417.2.2 uint8_t tempData_t::temperatureDataLen

8.417.2.3 uint32_t tempData_t::timeOfFirstSample

8.417.2.4 uint16_t tempData_t::timeOffset[64]

8.417.2.5 uint32_t tempData_t::timeSource

8.418 transferRouteMessageTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- sMSTransferRouteMTMessageInfo TransferRouteMTMessageInfo

8.418.1 Detailed Description

Parameters

TlvPresent	• Boolean indicating the presence of the TLV in the QMI response
TransferRouteMTMessageInfo	• Transfer Route MT Message • See sMSTransferRouteMTMessageInfo for more information

8.418.2 Field Documentation

8.418.2.1 uint8_t transferRouteMessageTlv::TlvPresent

8.418.2.2 sMSTransferRouteMTMessageInfo transferRouteMessageTlv::TransferRouteMTMessageInfo

8.419 transferStatInd Struct Reference

Data Fields

- uint8_t StatsPeriod
- uint32_t StatsMask

8.419.1 Detailed Description

Parameters

StatsPeriod	Field Period between transfer statistic reports.
StatsMask	requested statistic bit mask.

8.419.2 Field Documentation

8.419.2.1 uint32_t transferStatInd::StatsMask

8.419.2.2 uint8_t transferStatInd::StatsPeriod

8.420 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

8.420.1 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.

8.420.2 Field Documentation

8.420.2.1 uint8_t uim_appStatus::aidLength

8.420.2.2 uint8_t uim_appStatus::aidVal[255]

8.420.2.3 uint8_t uim_appStatus::appState

8.420.2.4 uint8_t uim_appStatus::appType

8.420.2.5 uint8_t uim_appStatus::persoFeature

8.420.2.6 uint8_t uim_appStatus::persoRetries

8.420.2.7 uint8_t uim_appStatus::persoState

8.420.2.8 uint8_t uim_appStatus::persoUnblockRetries

8.420.2.9 uint8_t uim_appStatus::pin1Retries

8.420.2.10 uint8_t uim_appStatus::pin1State

8.420.2.11 uint8_t uim_appStatus::pin2Retries

8.420.2.12 uint8_t uim_appStatus::pin2State

8.420.2.13 uint8_t uim_appStatus::puk1Retries

8.420.2.14 uint8_t uim_appStatus::puk2Retries

8.420.2.15 uint8_t uim_appStatus::univPin

8.421 uim_authenticateResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [1024]

8.421.1 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.

8.421.2 Field Documentation

8.421.2.1 uint8_t uim_authenticateResult::content[1024]

8.421.2.2 uint16_t uim_authenticateResult::contentLen

8.422 uim_authenticationData Struct Reference

Data Fields

- uint8_t context
- uint16_t dataLen
- uint8_t data [1024]

8.422.1 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_DESCRIPTION_LENGTH]	• Authenticate Data.

8.422.2 Field Documentation

8.422.2.1 uint8_t uim_authenticationData::context

8.422.2.2 uint8_t uim_authenticationData::data[1024]

8.422.2.3 uint16_t uim_authenticationData::dataLen

8.423 uim_cardResult Struct Reference

Data Fields

- uint8_t sw1
- uint8_t sw2

8.423.1 Detailed Description

This structure contains the information about the card result.

Parameters

sw1	• SW1 received from the card.
sw2	• SW2 received from the card.

8.423.2 Field Documentation

8.423.2.1 uint8_t uim_cardResult::sw1

8.423.2.2 uint8_t uim_cardResult::sw2

8.424 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]

8.424.1 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.

8.424.2 Field Documentation

8.424.2.1 uint16_t uim_cardStatus::index1xPri

8.424.2.2 uint16_t uim_cardStatus::index1xSec

8.424.2.3 uint16_t uim_cardStatus::indexGwPri

8.424.2.4 uint16_t uim_cardStatus::indexGwSec

8.424.2.5 uint8_t uim_cardStatus::numSlot

8.424.2.6 uim_slotInfo uim_cardStatus::SlotInfo[5]

8.425 uim_changeUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t oldPINLen
- uint8_t oldPINVal [255]
- uint8_t pinLen
- uint8_t pinVal [255]

8.425.1 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.

8.425.2 Field Documentation

8.425.2.1 uint8_t uim_changeUIMPIN::oldPINLen

8.425.2.2 uint8_t uim_changeUIMPIN::oldPINVal[255]

8.425.2.3 uint8_t uim_changeUIMPIN::pinID

8.425.2.4 uint8_t uim_changeUIMPIN::pinLen

8.425.2.5 uint8_t uim_changeUIMPIN::pinVal[255]

8.426 uim_depersonalizationInformation Struct Reference

Data Fields

- uint8_t feature
- uint8_t operation
- uint8_t ckLen
- uint8_t ckVal [255]

8.426.1 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.

8.426.2 Field Documentation

8.426.2.1 uint8_t uim_depersonalizationInformation::ckLen

8.426.2.2 uint8_t uim_depersonalizationInformation::ckVal[255]

8.426.2.3 uint8_t uim_depersonalizationInformation::feature

8.426.2.4 uint8_t uim_depersonalizationInformation::operation

8.427 uim_encryptedPIN1 Struct Reference

Data Fields

- uint8_t pin1Len
- uint8_t pin1Val [255]

8.427.1 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.

8.427.2 Field Documentation

8.427.2.1 uint8_t uim_encryptedPIN1::pin1Len

8.427.2.2 uint8_t uim_encryptedPIN1::pin1Val[255]

8.428 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]

8.428.1 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.

8.428.2 Field Documentation

8.428.2.1 uint16_t uim_fileAttributes::fileID

8.428.2.2 uint16_t uim_fileAttributes::fileSize

8.428.2.3 uint8_t uim_fileAttributes::fileType

8.428.2.4 uint16_t uim_fileAttributes::rawLen

8.428.2.5 uint8_t uim_fileAttributes::rawValue[255]

8.428.2.6 uint16_t uim_fileAttributes::recordCount

8.428.2.7 uint16_t uim_fileAttributes::recordSize

8.428.2.8 uint8_t uim_fileAttributes::secActivate

8.428.2.9 uint16_t uim_fileAttributes::secActivateMask

8.428.2.10 uint8_t uim_fileAttributes::secDeactivate

8.428.2.11 uint16_t uim_fileAttributes::secDeactivateMask

8.428.2.12 uint8_t uim_fileAttributes::secIncrease

8.428.2.13 uint16_t uim_fileAttributes::secIncreaseMask

8.428.2.14 uint8_t uim_fileAttributes::secRead

8.428.2.15 uint16_t uim_fileAttributes::secReadMask

8.428.2.16 uint8_t uim_fileAttributes::secWrite

8.428.2.17 uint16_t uim_fileAttributes::secWriteMask

8.429 uim_fileInfo Struct Reference

Data Fields

- uint16_t fileID
- uint8_t pathLen
- uint16_t path [255]

8.429.1 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).

8.429.2 Field Documentation

8.429.2.1 uint16_t uim_fileInfo::fileID

8.429.2.2 uint16_t uim_fileInfo::path[255]

8.429.2.3 uint8_t uim_fileInfo::pathLen

8.430 uim_hotSwapStatus Struct Reference

Data Fields

- uint8_t hotSwapLength
- uint8_t hotSwap [255]

8.430.1 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

8.430.2 Field Documentation

8.430.2.1 uint8_t uim_hotSwapStatus::hotSwap[255]

8.430.2.2 uint8_t uim_hotSwapStatus::hotSwapLength

8.431 uim_personalizationStatus Struct Reference

Data Fields

- uint8_t numFeatures
- uint8_t feature [12]
- uint8_t verifyLeft [12]
- uint8_t unblockLeft [12]

8.431.1 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.

8.431.2 Field Documentation

8.431.2.1 uint8_t uim_personalizationStatus::feature[12]

8.431.2.2 uint8_t uim_personalizationStatus::numFeatures

8.431.2.3 uint8_t uim_personalizationStatus::unblockLeft[12]

8.431.2.4 uint8_t uim_personalizationStatus::verifyLeft[12]

8.432 uim_readResult Struct Reference

Data Fields

- uint16_t contentLen
- uint8_t content [255]

8.432.1 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.

8.432.2 Field Documentation

8.432.2.1 uint8_t uim_readResult::content[255]

8.432.2.2 uint16_t uim_readResult::contentLen

8.433 uim_readTransparentInfo Struct Reference

Data Fields

- uint16_t offset
- uint16_t length

8.433.1 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.

8.433.2 Field Documentation

8.433.2.1 uint16_t uim_readTransparentInfo::length

8.433.2.2 uint16_t uim_readTransparentInfo::offset

8.434 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]

8.434.1 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

8.434.2 Field Documentation

8.434.2.1 uint8_t uim_refreshevent::aid[255]

8.434.2.2 uint8_t uim_refreshevent::aidLength

8.434.2.3 uim_fileInfo uim_refreshevent::arrfileInfo[255]

8.434.2.4 uint8_t uim_refreshevent::mode

8.434.2.5 uint16_t uim_refreshevent::numOfFiles

8.434.2.6 uint8_t uim_refreshevent::sessionType

8.434.2.7 uint8_t uim_refreshevent::stage

8.435 uim_registerRefresh Struct Reference

Data Fields

- uint8_t registerFlag
- uint8_t voteForInit
- uint16_t numFiles
- uim_fileInfo arrfileInfo [255]

8.435.1 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

8.435.2 Field Documentation

8.435.2.1 uim_fileInfo uim_registerRefresh::arrfileInfo[255]

8.435.2.2 uint16_t uim_registerRefresh::numFiles

8.435.2.3 uint8_t uim_registerRefresh::registerFlag

8.435.2.4 uint8_t uim_registerRefresh::voteForInit

8.436 uim_remainingRetries Struct Reference

Data Fields

- uint8_t verifyLeft
- uint8_t unblockLeft

8.436.1 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.

8.436.2 Field Documentation

8.436.2.1 uint8_t uim_remainingRetries::unblockLeft

8.436.2.2 uint8_t uim_remainingRetries::verifyLeft

8.437 uim_sessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.437.1 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.

8.437.2 Field Documentation

8.437.2.1 uint8_t uim_sessionInformation::aid[255]

8.437.2.2 uint8_t uim_sessionInformation::aidLength

8.437.2.3 uint8_t uim_sessionInformation::sessionType

8.438 uim_setPINProtection Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinOperation

- uint8_t pinLength
- uint8_t pinValue [255]

8.438.1 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.

8.438.2 Field Documentation

8.438.2.1 uint8_t uim_setPINProtection::pinID

8.438.2.2 uint8_t uim_setPINProtection::pinLength

8.438.2.3 uint8_t uim_setPINProtection::pinOperation

8.438.2.4 uint8_t uim_setPINProtection::pinValue[255]

8.439 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]

8.439.1 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.

8.439.2 Field Documentation

8.439.2.1 uim_appStatus uim_slotInfo::AppStatus[10]

8.439.2.2 uint8_t uim_slotInfo::cardState

8.439.2.3 uint8_t uim_slotInfo::errorState

8.439.2.4 uint8_t uim_slotInfo::numApp

8.439.2.5 uint8_t uim_slotInfo::upinRetries

8.439.2.6 uint8_t uim_slotInfo::upinState

8.439.2.7 uint8_t uim_slotInfo::upukRetries

8.440 uim_UIMSessionInformation Struct Reference

Data Fields

- uint8_t sessionType
- uint8_t aidLength
- uint8_t aid [255]

8.440.1 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.

8.440.2 Field Documentation

8.440.2.1 uint8_t uim_UIMSessionInformation::aid[255]

8.440.2.2 uint8_t uim_UIMSessionInformation::aidLength

8.440.2.3 uint8_t uim_UIMSessionInformation::sessionType

8.441 uim_unblockUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pukLen
- uint8_t pukVal [255]
- uint8_t newPINLen
- uint8_t newPINVal [255]

8.441.1 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.

8.441.2 Field Documentation

8.441.2.1 uint8_t uim_unblockUIMPIN::newPINLen

8.441.2.2 uint8_t uim_unblockUIMPIN::newPINVal[255]

8.441.2.3 uint8_t uim_unblockUIMPIN::pinID

8.441.2.4 uint8_t uim_unblockUIMPIN::pukLen

8.441.2.5 uint8_t uim_unblockUIMPIN::pukVal[255]

8.442 uim_verifyUIMPIN Struct Reference

Data Fields

- uint8_t pinID
- uint8_t pinLen
- uint8_t pinVal [255]

8.442.1 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.

8.442.2 Field Documentation

8.442.2.1 uint8_t uim_verifyUIMPIN::pinID

8.442.2.2 uint8_t uim_verifyUIMPIN::pinLen

8.442.2.3 uint8_t uim_verifyUIMPIN::pinVal[255]

8.443 unpack_dms_GetActivationState_t Struct Reference

Data Fields

- uint8_t state

8.443.1 Detailed Description

Parameters

pActivation-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

8.443.2 Field Documentation

8.443.2.1 uint8_t unpack_dms_GetActivationState_t::state

8.444 unpack_dms_GetBandCapability_t Struct Reference

Data Fields

- uint32_t BandCapability
- uint16_t Tlvresult

8.444.1 Field Documentation

8.444.1.1 uint32_t unpack_dms_GetBandCapability_t::BandCapability

8.444.1.2 uint16_t unpack_dms_GetBandCapability_t::Tlvresult

8.445 unpack_dms_GetCrashAction_t Struct Reference

Data Fields

- uint8_t DevCrashState
- uint16_t Tlvresult

8.445.1 Field Documentation

8.445.1.1 uint8_t unpack_dms_GetCrashAction_t::DevCrashState

8.445.1.2 uint16_t unpack_dms_GetCrashAction_t::Tlvresult

8.446 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

8.446.1 Field Documentation

8.446.1.1 uint8_t unpack_dms_GetCustFeature_t::DHCPRelayEnabled

8.446.1.2 uint8_t unpack_dms_GetCustFeature_t::DisableIMSI

8.446.1.3 uint32_t unpack_dms_GetCustFeature_t::GpsEnable

8.446.1.4 uint8_t unpack_dms_GetCustFeature_t::GPSLPM

8.446.1.5 uint8_t unpack_dms_GetCustFeature_t::GPSSel

8.446.1.6 uint16_t unpack_dms_GetCustFeature_t::IPFamSupport

8.446.1.7 uint8_t unpack_dms_GetCustFeature_t::IsVoiceEnabled

8.446.1.8 uint8_t unpack_dms_GetCustFeature_t::RMAutoConnect

8.446.1.9 uint8_t unpack_dms_GetCustFeature_t::SMSSupport

8.446.1.10 uint16_t unpack_dms_GetCustFeature_t::Tlvresult

8.447 unpack_dms_GetCustFeaturesV2_t Struct Reference

Data Fields

- DMSgetCustomFeatureV2 GetCustomFeatureV2
- uint16_t Tlvresult

8.447.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

Tlvresult	• Unpack Result

8.447.2 Field Documentation

8.447.2.1 DMSgetCustomFeatureV2 unpack_dms_GetCustFeaturesV2_t::GetCustomFeatureV2

8.447.2.2 uint16_t unpack_dms_GetCustFeaturesV2_t::Tlvresult

8.448 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

8.448.1 Field Documentation

8.448.1.1 uint32_t unpack_dms_GetDeviceCap_t::DataServiceCapability

8.448.1.2 uint32_t unpack_dms_GetDeviceCap_t::MaxRXChannelRate

8.448.1.3 uint32_t unpack_dms_GetDeviceCap_t::MaxTXChannelRate

8.448.1.4 uint8_t unpack_dms_GetDeviceCap_t::RadioIfaces[64]

8.448.1.5 uint32_t unpack_dms_GetDeviceCap_t::RadioIfacesSize

8.448.1.6 uint32_t unpack_dms_GetDeviceCap_t::SimCapability

8.448.1.7 uint16_t unpack_dms_GetDeviceCap_t::Tlvresult

8.449 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]

8.449.1 Detailed Description

Parameters

maxTxChannel-Rate	Maximum Tx transmission rate in bits per second.
maxRxChannel-Rate	Maximum Rx transmission rate in bits per second
dataServiceCa-Capability	data service capability
simCapability	SIM Capability
radioIfacesSize	radio interface length
RadioIfaces	radio interfaces

8.449.2 Field Documentation

8.449.2.1 uint32_t unpack_dms_GetDeviceCapabilities_t::dataServiceCaCapability

8.449.2.2 uint32_t unpack_dms_GetDeviceCapabilities_t::maxRxChannelRate

8.449.2.3 uint32_t unpack_dms_GetDeviceCapabilities_t::maxTxChannelRate

8.449.2.4 uint8_t unpack_dms_GetDeviceCapabilities_t::RadioIfaces[255]

8.449.2.5 uint32_t unpack_dms_GetDeviceCapabilities_t::radioIfacesSize

8.449.2.6 uint32_t unpack_dms_GetDeviceCapabilities_t::simCapability

8.450 unpack_dms_GetDeviceHardwareRev_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t Tlvresult

8.450.1 Field Documentation

8.450.1.1 char unpack_dms_GetDeviceHardwareRev_t::String[255]

8.450.1.2 uint8_t unpack_dms_GetDeviceHardwareRev_t::stringSize

8.450.1.3 uint16_t unpack_dms_GetDeviceHardwareRev_t::Tlvresult

8.451 unpack_dms_GetDeviceMfr_t Struct Reference

Data Fields

- uint8_t stringSize
- char String [255]
- uint16_t Tlvresult

8.451.1 Field Documentation

8.451.1.1 char unpack_dms_GetDeviceMfr_t::String[255]

8.451.1.2 uint8_t unpack_dms_GetDeviceMfr_t::stringSize

8.451.1.3 uint16_t unpack_dms_GetDeviceMfr_t::Tlvresult

8.452 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

8.452.1 Field Documentation

8.452.1.1 uint8_t unpack_dms_GetDeviceSerialNumbers_t::esnSize

8.452.1.2 char unpack_dms_GetDeviceSerialNumbers_t::ESNString[255]

8.452.1.3 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSize

8.452.1.4 char unpack_dms_GetDeviceSerialNumbers_t::IMEIString[255]

8.452.1.5 uint8_t unpack_dms_GetDeviceSerialNumbers_t::imeiSvnSize

8.452.1.6 char unpack_dms_GetDeviceSerialNumbers_t::ImeiSvnString[255]

8.452.1.7 uint8_t unpack_dms_GetDeviceSerialNumbers_t::meidSize

8.452.1.8 char unpack_dms_GetDeviceSerialNumbers_t::MEIDString[255]

8.452.1.9 uint16_t unpack_dms_GetDeviceSerialNumbers_t::Tlvresult

8.453 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

8.453.1 Detailed Description

Parameters

modelid_str	Mode ID String.
bootversion_str	Boot Version.
appversion_str	Application Version String.
sku_str	SKU String.
packageid_str	• Package ID String. • deprecated on EM/MC74xx(9x30) devices

carrier_str	Carrier String.
priversion_str	PRI Version String.
priversion_str	PRI Version String.
cur_carr_name	Current Carrier Name String.
cur_carr_rev	Current Carrier Revision String.
Tlvresult	Tlv Result.

8.453.2 Field Documentation

8.453.2.1 char unpack_dms_GetFirmwareInfo_t::appversion_str[85]

8.453.2.2 char unpack_dms_GetFirmwareInfo_t::bootversion_str[85]

8.453.2.3 char unpack_dms_GetFirmwareInfo_t::carrier_str[20]

8.453.2.4 char unpack_dms_GetFirmwareInfo_t::cur_carr_name[17]

8.453.2.5 char unpack_dms_GetFirmwareInfo_t::cur_carr_rev[13]

8.453.2.6 char unpack_dms_GetFirmwareInfo_t::modelid_str[20]

8.453.2.7 char unpack_dms_GetFirmwareInfo_t::packageid_str[85]

8.453.2.8 char unpack_dms_GetFirmwareInfo_t::priversion_str[16]

8.453.2.9 char unpack_dms_GetFirmwareInfo_t::sku_str[15]

8.453.2.10 uint16_t unpack_dms_GetFirmwareInfo_t::Tlvresult

8.454 unpack_dms_GetFirmwareRevision_t Struct Reference

Data Fields

- uint8_t amssSize
- char AMSSString [255]
- char PRIString [255]
- uint16_t Tlvresult

8.454.1 Field Documentation

8.454.1.1 uint8_t unpack_dms_GetFirmwareRevision_t::amssSize

8.454.1.2 char unpack_dms_GetFirmwareRevision_t::AMSSString[255]

8.454.1.3 char unpack_dms_GetFirmwareRevision_t::PRIString[255]

8.454.1.4 uint16_t unpack_dms_GetFirmwareRevision_t::Tlvresult

8.455 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

8.455.1 Detailed Description

Parameters

amssstring	AMSS revision string
bootstring	boot code revision string
pristring	PRI revision string

8.455.2 Field Documentation

8.455.2.1 uint8_t unpack_dms_GetFirmwareRevisions_t::amssSize

8.455.2.2 char unpack_dms_GetFirmwareRevisions_t::AMSSString[255]

8.455.2.3 uint8_t unpack_dms_GetFirmwareRevisions_t::bootSize

8.455.2.4 char unpack_dms_GetFirmwareRevisions_t::BootString[255]

8.455.2.5 uint8_t unpack_dms_GetFirmwareRevisions_t::priSize

8.455.2.6 char unpack_dms_GetFirmwareRevisions_t::PRIString[255]

8.455.2.7 uint16_t unpack_dms_GetFirmwareRevisions_t::Tlvresult

8.456 unpack_dms_GetFSN_t Struct Reference

Data Fields

- char String [255]
- uint16_t Tlvresult

8.456.1 Field Documentation

8.456.1.1 char unpack_dms_GetFSN_t::String[255]

8.456.1.2 uint16_t unpack_dms_GetFSN_t::Tlvresult

8.457 unpack_dms_GetHardwareRevision_t Struct Reference

Data Fields

- char hwVer [255]

8.457.1 Detailed Description

Parameters

hwVer	hardware vesion

8.457.2 Field Documentation

8.457.2.1 char unpack_dms_GetHardwareRevision_t::hwVer[255]

8.458 unpack_dms_GetIMSI_t Struct Reference

Data Fields

- char imsi [255]
- uint16_t Tlvresult

8.458.1 Field Documentation

8.458.1.1 char unpack_dms_GetIMSI_t::imsi[255]

8.458.1.2 uint16_t unpack_dms_GetIMSI_t::Tlvresult

8.459 unpack_dms_GetManufacturer_t Struct Reference

Data Fields

- char manufacturer [255]
- uint16_t Tlvresult

8.459.1 Detailed Description

This structure is used to store device manufacturer information.

Parameters

manufacturer[OUT]	• NULL terminated string
Tlvresult	• Unpack Result

8.459.2 Field Documentation

8.459.2.1 char unpack_dms_GetManufacturer_t::manufacturer[255]

8.459.2.2 uint16_t unpack_dms_GetManufacturer_t::Tlvresult

8.460 unpack_dms_GetModelID_t Struct Reference

Data Fields

- char modelid [255]

- uint16_t Tlvresult

8.460.1 Detailed Description

Parameters

modelid	device model id

8.460.2 Field Documentation

8.460.2.1 char unpack_dms_GetModelID_t::modelid[255]

8.460.2.2 uint16_t unpack_dms_GetModelID_t::Tlvresult

8.461 unpack_dms_GetNetworkTime_t Struct Reference

Data Fields

- uint16_t source
- uint64_t timestamp
- uint16_t Tlvresult

8.461.1 Detailed Description

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.

8.461.2 Field Documentation

8.461.2.1 uint16_t unpack_dms_GetNetworkTime_t::source

8.461.2.2 uint64_t unpack_dms_GetNetworkTime_t::timestamp

8.461.2.3 uint16_t unpack_dms_GetNetworkTime_t::Tlvresult

8.462 unpack_dms_GetOfflineReason_t Struct Reference

Data Fields

- uint32_t * pReasonMask
- uint32_t * pbPlatform
- uint16_t Tlvresult

8.462.1 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

8.462.2 Field Documentation

8.462.2.1 uint32_t* unpack_dms_GetOfflineReason_t::pbPlatform

8.462.2.2 uint32_t* unpack_dms_GetOfflineReason_t::pReasonMask

8.462.2.3 uint16_t unpack_dms_GetOfflineReason_t::Tlvresult

8.463 unpack_dms_GetPower_t Struct Reference

Data Fields

- uint32_t OperationMode
- uint32_t OfflineReason
- uint32_t HardwareControlledMode
- uint16_t Tlvresult

8.463.1 Detailed Description

Parameters

OperationMode	operating mode
OfflineReason	offline reason
Hardware-ControlledMode	hardware restricted mode

8.463.2 Field Documentation

8.463.2.1 uint32_t unpack_dms_GetPower_t::HardwareControlledMode

8.463.2.2 uint32_t unpack_dms_GetPower_t::OfflineReason

8.463.2.3 uint32_t unpack_dms_GetPower_t::OperationMode

8.463.2.4 uint16_t unpack_dms_GetPower_t::Tlvresult

8.464 unpack_dms_GetPRLVersion_t Struct Reference

Data Fields

- uint8_t u8PRLPreference
- uint16_t u16PRLVersion
- uint16_t Tlvresult

8.464.1 Field Documentation

8.464.1.1 uint16_t unpack_dms_GetPRLVersion_t::Tlvresult

8.464.1.2 uint16_t unpack_dms_GetPRLVersion_t::u16PRLVersion

8.464.1.3 uint8_t unpack_dms_GetPRLVersion_t::u8PRLPreference

8.465 unpack_dms_GetSerialNumbers_t Struct Reference

Data Fields

- char esn [255]
- char imei_no [255]
- char meid [255]
- char imeisv_svn [255]

8.465.1 Detailed Description

Parameters

esn	Electronic Serial Number of the device
imei_no	International Mobile Equipment Identity of the device.
meid	Mobile Equipment Identifier of the device.
imeisv_svn	imei software version revision

8.465.2 Field Documentation

8.465.2.1 char unpack_dms_GetSerialNumbers_t::esn[255]

8.465.2.2 char unpack_dms_GetSerialNumbers_t::imei_no[255]

8.465.2.3 char unpack_dms_GetSerialNumbers_t::imeisv_svn[255]

8.465.2.4 char unpack_dms_GetSerialNumbers_t::meid[255]

8.466 unpack_dms_GetUSBComp_t Struct Reference

Data Fields

- uint8_t USBComp [255]
- uint8_t NumSupUSBComps
- uint8_t SupUSBComps
- uint16_t Tlvresult

8.466.1 Field Documentation

8.466.1.1 uint8_t unpack_dms_GetUSBComp_t::NumSupUSBComps

8.466.1.2 uint8_t unpack_dms_GetUSBComp_t::SupUSBComps

8.466.1.3 uint16_t unpack_dms_GetUSBComp_t::Tlvresult

8.466.1.4 uint8_t unpack_dms_GetUSBComp_t::USBComp[255]

8.467 unpack_dms_GetVoiceNumber_t Struct Reference

Data Fields

- uint8_t voiceNumberSize
- char VoiceNumber [255]
- uint8_t minSize
- char MIN [255]
- uint16_t Tlvresult

8.467.1 Field Documentation

8.467.1.1 char unpack_dms_GetVoiceNumber_t::MIN[255]

8.467.1.2 uint8_t unpack_dms_GetVoiceNumber_t::minSize

8.467.1.3 uint16_t unpack_dms_GetVoiceNumber_t::Tlvresult

8.467.1.4 char unpack_dms_GetVoiceNumber_t::VoiceNumber[255]

8.467.1.5 uint8_t unpack_dms_GetVoiceNumber_t::voiceNumberSize

8.468 unpack_dms_ResetToFactoryDefaults_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.468.1 Detailed Description

This structure contains reset to factory default unpack

Parameters

Tlvresult	• Unpack Result

8.468.2 Field Documentation

8.468.2.1 uint16_t unpack_dms_ResetToFactoryDefaults_t::Tlvresult

8.469 unpack_dms_SetActivationStatusCallback_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.469.1 Detailed Description

unpack structure for Activation status callback

Parameters

Tlvresult	• Unpack Result

8.469.2 Field Documentation

8.469.2.1 uint16_t unpack_dms_SetActivationStatusCallback_t::Tlvresult

8.470 unpack_dms_SetCrashAction_t Struct Reference

Data Fields

- uint8_t notused

8.470.1 Detailed Description

Modem response. Not used

8.470.2 Field Documentation

8.470.2.1 uint8_t unpack_dms_SetCrashAction_t::notused

8.471 unpack_dms_SetCustFeature_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.471.1 Field Documentation

8.471.1.1 uint16_t unpack_dms_SetCustFeature_t::Tlvresult

8.472 unpack_dms_SetCustFeaturesV2_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.472.1 Detailed Description

This structure contains customization settings set to modem unpack

Parameters

Tlvresult	• Unpack Result

8.472.2 Field Documentation

8.472.2.1 uint16_t unpack_dms_SetCustFeaturesV2_t::Tlvresult

8.473 unpack_dms_SetEventReport_ind_t Struct Reference

Data Fields

- dms_ActivationStatusTlv ActivationStatusTlv
- dms_OperatingModeTlv OperatingModeTlv
- uint16_t Tlvresult

8.473.1 Detailed Description

DMS Event Report indication structure

Parameters

ActivationStatus-Tlv	• See dms_ActivationStatusTlv
OperatingMode-Tlv	• See dms_OperatingModeTlv

Tlvresult	• Unpack Result

8.473.2 Field Documentation

8.473.2.1 dms_ActivationStatusTlv unpack_dms_SetEventReport_ind_t::ActivationStatusTlv

8.473.2.2 dms_OperatingModeTlv unpack_dms_SetEventReport_ind_t::OperatingModeTlv

8.473.2.3 uint16_t unpack_dms_SetEventReport_ind_t::Tlvresult

8.474 unpack_dms_SetEventReport_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.474.1 Field Documentation

8.474.1.1 uint16_t unpack_dms_SetEventReport_t::Tlvresult

8.475 unpack_dms_SetFirmwarePreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.475.1 Field Documentation

8.475.1.1 uint16_t unpack_dms_SetFirmwarePreference_t::Tlvresult

8.476 unpack_dms_SetPower_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.476.1 Field Documentation

8.476.1.1 uint16_t unpack_dms_SetPower_t::Tlvresult

8.477 unpack_dms_SetUSBComp_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.477.1 Field Documentation

8.477.1.1 uint16_t unpack_dms_SetUSBComp_t::Tlvresult

8.478 unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult

8.478.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

OUT]	type[OUT] • type of reset or power down, possible values listed below: – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down
OUT]	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

8.478.2 Field Documentation

8.478.2.1 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::source

8.478.2.2 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::Tlvresult

8.478.2.3 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t::type

8.479 unpack_dms_SLQSDmsSwiGetResetInfo_t Struct Reference

Data Fields

- uint8_t type
- uint8_t source
- uint16_t Tlvresult

8.479.1 Detailed Description

This structure contains the TLV required to Get Reset Info.

Parameters

OUT]	type[OUT] • type of reset or power down, possible values listed below: – 0 - unknown – 1 - warm – 2 - hard – 3 - crash – 4 - power down
OUT]	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

8.479.2 Field Documentation

8.479.2.1 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::source

8.479.2.2 uint16_t unpack_dms_SLQSDmsSwiGetResetInfo_t::Tlvresult

8.479.2.3 uint8_t unpack_dms_SLQSDmsSwiGetResetInfo_t::type

8.480 unpack_dms_SLQSDmsSwiIndicationRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.480.1 Detailed Description

This structure contains set registration state for different indication unpack

Parameters

Tlvresult	• Unpack Result

8.480.2 Field Documentation

8.480.2.1 uint16_t unpack_dms_SLQSDmsSwiIndicationRegister_t::Tlvresult

8.481 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

8.481.1 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

8.481.2 Field Documentation

8.481.2.1 uint64_t unpack_dms_SLQSGetBandCapability_t::bandCapability

8.481.2.2 int unpack_dms_SLQSGetBandCapability_t::is_LteBandCapability_Available

8.481.2.3 int unpack_dms_SLQSGetBandCapability_t::is_TdsBandCapability_Available

8.481.2.4 uint64_t unpack_dms_SLQSGetBandCapability_t::LteBandCapability

8.481.2.5 uint64_t unpack_dms_SLQSGetBandCapability_t::TdsBandCapability

8.482 unpack_dms_SLQSGetERIFile_t Struct Reference

Data Fields

- uint16_t Tlvresult
- eriDataparams eriFile

8.482.1 Detailed Description

This structure contains Get ERI file unpack

Parameters

Tlvresult	• Unpack Result

8.482.2 Field Documentation

8.482.2.1 eriDataparams unpack_dms_SLQSGetERIFile_t::eriFile

8.482.2.2 uint16_t unpack_dms_SLQSGetERIFile_t::Tlvresult

8.483 unpack_dms_SLQSSwiClearDyingGaspStatistics_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.483.1 Detailed Description

This structure contains Clear Dying GASP unpack

Parameters

Tlvresult	• Unpack Result

8.483.2 Field Documentation

8.483.2.1 uint16_t unpack_dms_SLQSSwiClearDyingGaspStatistics_t::Tlvresult

8.484 unpack_dms_SLQSSwiGetCrashInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult
- crashInfoParams crashInfoParam

8.484.1 Detailed Description

This structure contains SWI get crash information unpack information

Parameters

Tlvresult	• Unpack Result
crashInfoParam[-OUT]	• See crashInfoParams

8.484.2 Field Documentation

8.484.2.1 crashInfoParams unpack_dms_SLQSSwiGetCrashInfo_t::crashInfoParam

8.484.2.2 uint16_t unpack_dms_SLQSSwiGetCrashInfo_t::Tlvresult

8.485 unpack_dms_SLQSSwiGetDyingGaspCfg_t Struct Reference

Data Fields

- packgetDyingGaspCfg * pGetDyingGaspCfg
- uint16_t Tlvresult

8.485.1 Detailed Description

This structure contains Get Dying GASP Config unpack

Parameters

Tlvresult	• Unpack Result

8.485.2 Field Documentation

8.485.2.1 packgetDyingGaspCfg∗ unpack_dms_SLQSSwiGetDyingGaspCfg_t::pGetDyingGaspCfg

8.485.2.2 uint16_t unpack_dms_SLQSSwiGetDyingGaspCfg_t::Tlvresult

8.486 unpack_dms_SLQSSwiGetDyingGaspStatistics_t Struct Reference

Data Fields

- packgetDyingGaspStatistics ∗ pGetDyingGaspStatistics
- uint16_t Tlvresult

8.486.1 Detailed Description

This structure contains Get Dying GASP Statistics.

Parameters

Tlvresult	• Unpack Result

8.486.2 Field Documentation

8.486.2.1 packgetDyingGaspStatistics∗ unpack_dms_SLQSSwiGetDyingGaspStatistics_t::pGetDyingGaspStatistics

8.486.2.2 uint16_t unpack_dms_SLQSSwiGetDyingGaspStatistics_t::Tlvresult

8.487 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]

8.487.1 Detailed Description

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

8.487.2 Field Documentation

8.487.2.1 char unpack_dms_SLQSSwiGetFirmwareCurr_t::carrier[16]

8.487.2.2 char unpack_dms_SLQSSwiGetFirmwareCurr_t::fwvers[16]

8.487.2.3 uint8_t unpack_dms_SLQSSwiGetFirmwareCurr_t::numEntries

8.487.2.4 image_info_t* unpack_dms_SLQSSwiGetFirmwareCurr_t::pCurrImgInfo

8.487.2.5 char unpack_dms_SLQSSwiGetFirmwareCurr_t::pkgver[16]

8.487.2.6 char unpack_dms_SLQSSwiGetFirmwareCurr_t::priver[16]

8.488 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

8.488.1 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 $\sim$ 0FF - Insignia defined error code * 100 $\sim$ 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

8.488.2 Field Documentation

8.488.2.1 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::imgType

8.488.2.2 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::logString[255]

8.488.2.3 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refData

8.488.2.4 uint8_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::refString[15]

8.488.2.5 uint32_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::ResCode

8.488.2.6 uint16_t unpack_dms_SLQSSwiGetFwUpdateStatus_t::TlvResult

8.489 unpack_dms_SLQSSwiGetHostDevInfo_t Struct Reference

Data Fields

- uint16_t TlvResult
- char manString [255]
- char modelString [255]
- char swVerString [255]
- char plasmaIDString [255]

8.489.1 Detailed Description

This structure contains SWI get host device info unpack information

Parameters

TlvResult	• unpack Result
manString[OUT]	• optional parameter, host device manufacture
modelString[OUT]	• optional parameter, host device model
swVerString[OUT]	• optional parameter, host device software version
plasmaIDString[OUT]	• optional parameter, host device plasma ID

8.489.2 Field Documentation

8.489.2.1 char unpack_dms_SLQSSwiGetHostDevInfo_t::manString[255]

8.489.2.2 char unpack_dms_SLQSSwiGetHostDevInfo_t::modelString[255]

8.489.2.3 char unpack_dms_SLQSSwiGetHostDevInfo_t::plasmaIDString[255]

8.489.2.4 char unpack_dms_SLQSSwiGetHostDevInfo_t::swVerString[255]

8.489.2.5 uint16_t unpack_dms_SLQSSwiGetHostDevInfo_t::TlvResult

8.490 unpack_dms_SLQSSwiGetOSInfo_t Struct Reference

Data Fields

- uint16_t TlvResult
- char nameString [255]
- char versionString [255]

8.490.1 Detailed Description

This structure contains SWI get host os info unpack information

Parameters

TlvResult	• pack Result
nameString[OUT]	• optional parameter, host operating system name
VersionString[OUT]	• optional parameter, host operating system version

8.490.2 Field Documentation

8.490.2.1 char unpack_dms_SLQSSwiGetOSInfo_t::nameString[255]

8.490.2.2 uint16_t unpack_dms_SLQSSwiGetOSInfo_t::TlvResult

8.490.2.3 char unpack_dms_SLQSSwiGetOSInfo_t::versionString[255]

8.491 unpack_dms_SLQSSwiGetSerialNoExt_t Struct Reference

Data Fields

- uint16_t TlvResult
- char meidString [8]

8.491.1 Detailed Description

This structure contains SWI get device serial number extension unpack information

Parameters

TlvResult	• pack Result
meidString[OUT]	• optional parameter, mobile equipment identifier

8.491.2 Field Documentation

8.491.2.1 char unpack_dms_SLQSSwiGetSerialNoExt_t::meidString[8]

8.491.2.2 uint16_t unpack_dms_SLQSSwiGetSerialNoExt_t::TlvResult

8.492 unpack_dms_SLQSSwiSetDyingGaspCfg_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.492.1 Detailed Description

This structure contains set Dying GASP Config unpack

Parameters

Tlvresult	• Unpack Result

8.492.2 Field Documentation

8.492.2.1 uint16_t unpack_dms_SLQSSwiSetDyingGaspCfg_t::Tlvresult

8.493 unpack_dms_SLQSSwiSetHostDevInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.493.1 Detailed Description

This structure contains SWI set host dev info unpack

Parameters

Tlvresult	• Unpack Result

8.493.2 Field Documentation

8.493.2.1 uint16_t unpack_dms_SLQSSwiSetHostDevInfo_t::Tlvresult

8.494 unpack_dms_SLQSSwiSetOSInfo_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.494.1 Detailed Description

This structure contains SWI set host OS info unpack

Parameters

Tlvresult	• Unpack Result

8.494.2 Field Documentation

8.494.2.1 uint16_t unpack_dms_SLQSSwiSetOSInfo_t::Tlvresult

8.495 unpack_dms_SLQSUIMGetState_t Struct Reference

Data Fields

- uint8_t state
- uint16_t Tlvresult

8.495.1 Detailed Description

This structure contains UIM get state unpack information

Parameters

state[OUT]	• UIM state
Tlvresult	• Unpack Result

8.495.2 Field Documentation

8.495.2.1 uint8_t unpack_dms_SLQSUIMGetState_t::state

8.495.2.2 uint16_t unpack_dms_SLQSUIMGetState_t::Tlvresult

8.496 unpack_dms_UIMGetControlKeyStatus_t Struct Reference

Data Fields

- uint8_t facilityState
- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t Tlvresult

8.496.1 Detailed Description

This structure contains PIN retries status

Parameters

facilityState[OUT]	• Control key status – 0 - Deactivated – 1 - Activated – 2 - Blocked

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	• Unpack Result

8.496.2 Field Documentation

8.496.2.1 uint8_t unpack_dms_UIMGetControlKeyStatus_t::facilityState

8.496.2.2 uint16_t unpack_dms_UIMGetControlKeyStatus_t::Tlvresult

8.496.2.3 uint8_t unpack_dms_UIMGetControlKeyStatus_t::unblockRetriesLeft

8.496.2.4 uint8_t unpack_dms_UIMGetControlKeyStatus_t::verifyRetriesLeft

8.497 unpack_dms_UIMGetICCID_t Struct Reference

Data Fields

- uint8_t stringSize
- uint8_t String [255]
- uint16_t Tlvresult

8.497.1 Detailed Description

This structure contains Get ICCID pack

Parameters

stringSize	• Size of String.
String	• ICCID String.
Tlvresult	• Pack result.

8.497.2 Field Documentation

8.497.2.1 uint8_t unpack_dms_UIMGetICCID_t::String[255]

8.497.2.2 uint8_t unpack_dms_UIMGetICCID_t::stringSize

8.497.2.3 uint16_t unpack_dms_UIMGetICCID_t::Tlvresult

8.498 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

8.498.1 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

8.498.2 Field Documentation

8.498.2.1 uint8_t unpack_dms_UIMGetPINStatus_t::p1Status

8.498.2.2 uint8_t unpack_dms_UIMGetPINStatus_t::p1UnblockRetriesLeft

8.498.2.3 uint8_t unpack_dms_UIMGetPINStatus_t::p1VerifyRetriesLeft

8.498.2.4 uint8_t unpack_dms_UIMGetPINStatus_t::p2Status

8.498.2.5 uint8_t unpack_dms_UIMGetPINStatus_t::p2UnblockRetriesLeft

8.498.2.6 uint8_t unpack_dms_UIMGetPINStatus_t::p2VerifyRetriesLeft

8.498.2.7 uint16_t unpack_dms_UIMGetPINStatus_t::Tlvresult

8.499 unpack_dms_UIMSetControlKeyProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint16_t Tlvresult

8.499.1 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

8.499.2 Field Documentation

8.499.2.1 uint16_t unpack_dms_UIMSetControlKeyProtection_t::Tlvresult

8.499.2.2 uint8_t unpack_dms_UIMSetControlKeyProtection_t::verifyRetriesLeft

8.500 unpack_dms_UIMSetPINProtection_t Struct Reference

Data Fields

- uint8_t verifyRetriesLeft
- uint8_t unblockRetriesLeft
- uint16_t Tlvresult

8.500.1 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	• Unpack Result

8.500.2 Field Documentation

8.500.2.1 uint16_t unpack_dms_UIMSetPINProtection_t::Tlvresult

8.500.2.2 uint8_t unpack_dms_UIMSetPINProtection_t::unblockRetriesLeft

8.500.2.3 uint8_t unpack_dms_UIMSetPINProtection_t::verifyRetriesLeft

8.501 unpack_dms_UIMUnblockControlKey_t Struct Reference

Data Fields

- uint8_t unblockRetriesLeft
- uint16_t Tlvresult

8.501.1 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	• Unpack Result

8.501.2 Field Documentation

8.501.2.1 uint16_t unpack_dms_UIMUnblockControlKey_t::Tlvresult

8.501.2.2 uint8_t unpack_dms_UIMUnblockControlKey_t::unblockRetriesLeft

8.502 unpack_fms_GetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageListSize
- FMSPrefImageList * pImageList
- uint16_t Tlvresult

8.502.1 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

8.502.2 Field Documentation

8.502.2.1 uint32_t unpack_fms_GetImagesPreference_t::ImageListSize

8.502.2.2 FMSPrefImageList* unpack_fms_GetImagesPreference_t::pImageList

8.502.2.3 uint16_t unpack_fms_GetImagesPreference_t::Tlvresult

8.503 unpack_fms_GetStoredImages_t Struct Reference

Data Fields

- uint32_t imagelistSize
- FMSImageList imageList
- uint16_t TlvResult

8.503.1 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

8.503.2 Field Documentation

8.503.2.1 FMSImageList unpack_fms_GetStoredImages_t::imageList

8.503.2.2 uint32_t unpack_fms_GetStoredImages_t::imagelistSize

8.503.2.3 uint16_t unpack_fms_GetStoredImages_t::TlvResult

8.504 unpack_fms_SetImagesPreference_t Struct Reference

Data Fields

- uint32_t ImageTypesSize
- uint8_t ImageTypes [255]
- uint16_t TlvResult

8.504.1 Detailed Description

This structure contains the Set Images Preference unpack

Parameters

ImageTypesSize	• Image Type Size
ImageTypes	• Image Type

Tlvresult	• Unpack result

8.504.2 Field Documentation

8.504.2.1 uint8_t unpack_fms_SetImagesPreference_t::ImageTypes[255]

8.504.2.2 uint32_t unpack_fms_SetImagesPreference_t::ImageTypesSize

8.504.2.3 uint16_t unpack_fms_SetImagesPreference_t::Tlvresult

8.505 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 Tlvresult

8.505.1 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

8.505.2 Field Documentation

8.505.2.1 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtEllipsoid

8.505.2.2 uint32_t* unpack_loc_BestAvailPos_Ind_t::pAltitudeWrtMeanSeaLevel

8.505.2.3 loc_gpsTime* unpack_loc_BestAvailPos_Ind_t::pGpsTime

8.505.2.4 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeading

8.505.2.5 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHeadingUnc

8.505.2.6 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorCirConf

8.505.2.7 uint8_t* unpack_loc_BestAvailPos_Ind_t::pHorEllpConf

8.505.2.8 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorReliability

8.505.2.9 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncCircular

8.505.2.10 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseOrientAzimuth

8.505.2.11 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMajor

8.505.2.12 uint32_t* unpack_loc_BestAvailPos_Ind_t::pHorUncEllipseSemiMinor

8.505.2.13 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLatitude

8.505.2.14 uint64_t* unpack_loc_BestAvailPos_Ind_t::pLongitude

8.505.2.15 uint32_t* unpack_loc_BestAvailPos_Ind_t::pMagneticDeviation

8.505.2.16 loc_precisionDilution* unpack_loc_BestAvailPos_Ind_t::pPrecisionDilution

8.505.2.17 loc_sensorDataUsage* unpack_loc_BestAvailPos_Ind_t::pSensorDataUsage

8.505.2.18 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedHorizontal

8.505.2.19 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedUnc

8.505.2.20 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVertical

8.505.2.21 uint32_t* unpack_loc_BestAvailPos_Ind_t::pSpeedVerticalUnc

8.505.2.22 loc_svUsedforFix* unpack_loc_BestAvailPos_Ind_t::pSvUsedforFix

8.505.2.23 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTechnologyMask

8.505.2.24 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeSrc

8.505.2.25 uint64_t* unpack_loc_BestAvailPos_Ind_t::pTimestampUtc

8.505.2.26 uint32_t* unpack_loc_BestAvailPos_Ind_t::pTimeUnc

8.505.2.27 uint8_t* unpack_loc_BestAvailPos_Ind_t::pVertConfidence

8.505.2.28 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertReliability

8.505.2.29 uint32_t* unpack_loc_BestAvailPos_Ind_t::pVertUnc

8.505.2.30 uint32_t* unpack_loc_BestAvailPos_Ind_t::pXid

8.505.2.31 uint32_t unpack_loc_BestAvailPos_Ind_t::status

8.505.2.32 uint16_t unpack_loc_BestAvailPos_Ind_t::TlvResult

8.506 unpack_loc_Delete_Assist_Data_t Struct Reference

Data Fields

- uint16_t TlvResult

8.506.1 Detailed Description

This structure contains LOC delete assist data unpack

Parameters

TlvResult	• Unpack result.

8.506.2 Field Documentation

8.506.2.1 uint16_t unpack_loc_Delete_Assist_Data_t::TlvResult

8.507 unpack_loc_DeleteAssistData_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t TlvResult

8.507.1 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

8.507.2 Field Documentation

8.507.2.1 uint32_t unpack_loc_DeleteAssistData_Ind_t::status

8.507.2.2 uint16_t unpack_loc_DeleteAssistData_Ind_t::Tlvresult

8.508 unpack_loc_EngineState_Ind_t Struct Reference

Data Fields

- uint32_t engineState
- uint16_t Tlvresult

8.508.1 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

8.508.2 Field Documentation

8.508.2.1 uint32_t unpack_loc_EngineState_Ind_t::engineState

8.508.2.2 uint16_t unpack_loc_EngineState_Ind_t::Tlvresult

8.509 unpack_loc_EventRegister_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.509.1 Detailed Description

This structure contains Event Register unpack

Parameters

Tlvresult	• Unpack result.

8.509.2 Field Documentation

8.509.2.1 uint16_t unpack_loc_EventRegister_t::Tlvresult

8.510 unpack_loc_GnssSvInfo_Ind_t Struct Reference

Data Fields

- uint8_t altitudeAssumed
- loc_satelliteInfo * pSatelliteInfo
- uint16_t Tlvresult

8.510.1 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

8.510.2 Field Documentation

8.510.2.1 uint8_t unpack_loc_GnssSvInfo_Ind_t::altitudeAssumed

8.510.2.2 loc_satelliteInfo* unpack_loc_GnssSvInfo_Ind_t::pSatelliteInfo

8.510.2.3 uint16_t unpack_loc_GnssSvInfo_Ind_t::Tlvresult

8.511 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

8.511.1 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

8.511.2 Field Documentation

8.511.2.1 uint8_t* unpack_loc_PositionRpt_Ind_t::pAltitudeAssumed

8.511.2.2 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtEllipsoid

8.511.2.3 uint32_t* unpack_loc_PositionRpt_Ind_t::pAltitudeWrtMeanSeaLevel

8.511.2.4 uint32_t* unpack_loc_PositionRpt_Ind_t::pFixId

8.511.2.5 loc_gpsTime* unpack_loc_PositionRpt_Ind_t::pGpsTime

8.511.2.6 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeading

8.511.2.7 uint32_t* unpack_loc_PositionRpt_Ind_t::pHeadingUnc

8.511.2.8 uint8_t* unpack_loc_PositionRpt_Ind_t::pHorConfidence

8.511.2.9 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorReliability

8.511.2.10 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncCircular

8.511.2.11 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseOrientAzimuth

8.511.2.12 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMajor

8.511.2.13 uint32_t* unpack_loc_PositionRpt_Ind_t::pHorUncEllipseSemiMinor

8.511.2.14 uint64_t* unpack_loc_PositionRpt_Ind_t::pLatitude

8.511.2.15 uint8_t* unpack_loc_PositionRpt_Ind_t::pLeapSeconds

8.511.2.16 uint64_t* unpack_loc_PositionRpt_Ind_t::pLongitude

8.511.2.17 uint32_t* unpack_loc_PositionRpt_Ind_t::pMagneticDeviation

8.511.2.18 loc_precisionDilution* unpack_loc_PositionRpt_Ind_t::pPrecisionDilution

8.511.2.19 loc_sensorDataUsage* unpack_loc_PositionRpt_Ind_t::pSensorDataUsage

8.511.2.20 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedHorizontal

8.511.2.21 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedUnc

8.511.2.22 uint32_t* unpack_loc_PositionRpt_Ind_t::pSpeedVertical

8.511.2.23 loc_svUsedforFix* unpack_loc_PositionRpt_Ind_t::pSvUsedforFix

8.511.2.24 uint32_t* unpack_loc_PositionRpt_Ind_t::pTechnologyMask

8.511.2.25 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeSrc

8.511.2.26 uint64_t* unpack_loc_PositionRpt_Ind_t::pTimestampUtc

8.511.2.27 uint32_t* unpack_loc_PositionRpt_Ind_t::pTimeUnc

8.511.2.28 uint8_t* unpack_loc_PositionRpt_Ind_t::pVertConfidence

8.511.2.29 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertReliability

8.511.2.30 uint32_t* unpack_loc_PositionRpt_Ind_t::pVertUnc

8.511.2.31 uint8_t unpack_loc_PositionRpt_Ind_t::sessionId

8.511.2.32 uint32_t unpack_loc_PositionRpt_Ind_t::sessionStatus

8.511.2.33 uint16_t unpack_loc_PositionRpt_Ind_t::Tlvresult

8.512 unpack_loc_SetExtPowerConfig_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult

8.512.1 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

8.512.2 Field Documentation

8.512.2.1 uint32_t unpack_loc_SetExtPowerConfig_Ind_t::status

8.512.2.2 uint16_t unpack_loc_SetExtPowerConfig_Ind_t::Tlvresult

8.513 unpack_loc_SetExtPowerState_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.513.1 Detailed Description

This structure contains Set Ext Power State unpack

Parameters

Tlvresult	• Unpack result.

8.513.2 Field Documentation

8.513.2.1 uint16_t unpack_loc_SetExtPowerState_t::Tlvresult

8.514 unpack_loc_SetOperationMode_Ind_t Struct Reference

Data Fields

- uint32_t status
- uint16_t Tlvresult

8.514.1 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

8.514.2 Field Documentation

8.514.2.1 uint32_t unpack_loc_SetOperationMode_Ind_t::status

8.514.2.2 uint16_t unpack_loc_SetOperationMode_Ind_t::Tlvresult

8.515 unpack_loc_SetOperationMode_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.515.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

Tlvresult	• Unpack result.

8.515.2 Field Documentation

8.515.2.1 uint16_t unpack_loc_SetOperationMode_t::Tlvresult

8.516 unpack_loc_SLQSLOCGetBestAvailPos_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.516.1 Detailed Description

This structure contains Set Operation Mode unpack

Parameters

Tlvresult	• Unpack result.

8.516.2 Field Documentation

8.516.2.1 uint16_t unpack_loc_SLQSLOCGetBestAvailPos_t::Tlvresult

8.517 unpack_loc_Start_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.517.1 Detailed Description

This structure contains Start LOC unpack

Parameters

Tlvresult	• Unpack result.

8.517.2 Field Documentation

8.517.2.1 uint16_t unpack_loc_Start_t::Tlvresult

8.518 unpack_loc_Stop_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.518.1 Detailed Description

This structure contains Stop LOC unpack

Parameters

Tlvresult	• Unpack result.

8.518.2 Field Documentation

8.518.2.1 uint16_t unpack_loc_Stop_t::Tlvresult

8.519 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

8.519.1 Detailed Description

Parameters

SCI	slot cycle index
SCM	station class mark
RegHomeSID	register on home sid
RegForeignSID	register on foreign sid
RegForeignNID	register on foreign nid
ForceRev0	force header revision
CustomSCP	custom SCP
Protocol	protocol
Booadcast	broadcast
Application	application
Roaming	roaming

8.519.2 Field Documentation

8.519.2.1 uint32_t unpack_nas_GetCDMANetworkParameters_t::Application

8.519.2.2 uint32_t unpack_nas_GetCDMANetworkParameters_t::Broadcast

8.519.2.3 uint8_t unpack_nas_GetCDMANetworkParameters_t::CustomSCP

8.519.2.4 uint8_t unpack_nas_GetCDMANetworkParameters_t::ForceRev0

8.519.2.5 uint32_t unpack_nas_GetCDMANetworkParameters_t::Protocol

8.519.2.6 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignNID

8.519.2.7 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegForeignSID

8.519.2.8 uint8_t unpack_nas_GetCDMANetworkParameters_t::RegHomeSID

8.519.2.9 uint32_t unpack_nas_GetCDMANetworkParameters_t::Roaming

8.519.2.10 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCI

8.519.2.11 uint8_t unpack_nas_GetCDMANetworkParameters_t::SCM

8.520 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

8.520.1 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 – 0x04 - Unicode – 0x09 - GSM 7-bit default
nw2DescLen	• Network Description Length (3GPP2 only).
pNw2Name	• Network Name (3GPP2 only).

8.520.2 Field Documentation

8.520.2.1 uint8_t unpack_nas_GetHomeNetwork3GPP2_t::nameSize

8.520.2.2 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMCC

8.520.2.3 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pMNC

8.520.2.4 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pName

8.520.2.5 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNID

8.520.2.6 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescDisp

8.520.2.7 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescEnc

8.520.2.8 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2DescLen

8.520.2.9 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MCC

8.520.2.10 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2MNC

8.520.2.11 uint8_t* unpack_nas_GetHomeNetwork3GPP2_t::pNw2Name

8.520.2.12 uint16_t* unpack_nas_GetHomeNetwork3GPP2_t::pSID

8.521 unpack_nas_GetHomeNetwork_t Struct Reference

Data Fields

- uint16_t mcc
- uint16_t mnc
- char name [255]
- uint16_t sid
- uint16_t nid

8.521.1 Detailed Description

Parameters

mcc	mobile country code
mnc	mobile network code
name	network name or description
sid	home network system id only applies to cdma2000
nid	home network id Only applies to cdma2000

8.521.2 Field Documentation

8.521.2.1 uint16_t unpack_nas_GetHomeNetwork_t::mcc

8.521.2.2 uint16_t unpack_nas_GetHomeNetwork_t::mnc

8.521.2.3 char unpack_nas_GetHomeNetwork_t::name[255]

8.521.2.4 uint16_t unpack_nas_GetHomeNetwork_t::nid

8.521.2.5 uint16_t unpack_nas_GetHomeNetwork_t::sid

8.522 unpack_nas_GetNetworkPreference_t Struct Reference

Data Fields

- uint32_t ActiveTechPref
- uint32_t Duration
- uint32_t PersistentTechPref
- uint16_t Tlvresult

8.522.1 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
PersistentTechnologyPref[-OUT]	• Bit field representing persistent radio technology preference – Same representation as the pTechnologyPref parameter
Tlvresult	• unpack result

8.522.2 Field Documentation

8.522.2.1 uint32_t unpack_nas_GetNetworkPreference_t::ActiveTechPref

8.522.2.2 uint32_t unpack_nas_GetNetworkPreference_t::Duration

8.522.2.3 uint32_t unpack_nas_GetNetworkPreference_t::PersistentTechPref

8.522.2.4 uint16_t unpack_nas_GetNetworkPreference_t::Tlvresult

8.523 unpack_nas_GetRFInfo_t Struct Reference

Data Fields

- uint8_t instancesSize
- RFBandInfoElements RFBandInfoElements [255]

8.523.1 Detailed Description

Parameters

instancesSize	number of elements in RF info instances array.
RFBandInfo-Elements	RF info instances array

8.523.2 Field Documentation

8.523.2.1 uint8_t unpack_nas_GetRFInfo_t::instancesSize

8.523.2.2 RFBandInfoElements unpack_nas_GetRFInfo_t::RFBandInfoElements[255]

8.524 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]

8.524.1 Detailed Description

Parameters

Registration-State	registration state
CSDomain	CS domain
PSDomain	PS domain
RAN	radio access network
RadioIfacesSize	radio interface size
RadioIfaces	radio interface list
Roaming	romaing indicator
MCC	Mobile country code
MNC	Mobile network code
nameSize	network name size
Name	network name
DataCapsLen	data capabilities len
DataCap	data capabilities

8.524.2 Field Documentation

8.524.2.1 uint32_t unpack_nas_GetServingNetwork_t::CSDomain

8.524.2.2 uint8_t unpack_nas_GetServingNetwork_t::DataCaps[255]

8.524.2.3 uint8_t unpack_nas_GetServingNetwork_t::DataCapsLen

8.524.2.4 uint16_t unpack_nas_GetServingNetwork_t::MCC

8.524.2.5 uint16_t unpack_nas_GetServingNetwork_t::MNC

8.524.2.6 uint8_t unpack_nas_GetServingNetwork_t::Name[255]

8.524.2.7 uint8_t unpack_nas_GetServingNetwork_t::nameSize

8.524.2.8 uint32_t unpack_nas_GetServingNetwork_t::PSDomain

8.524.2.9 uint8_t unpack_nas_GetServingNetwork_t::RadioIfaces[255]

8.524.2.10 uint8_t unpack_nas_GetServingNetwork_t::RadioIfacesSize

8.524.2.11 uint32_t unpack_nas_GetServingNetwork_t::RAN

8.524.2.12 uint32_t unpack_nas_GetServingNetwork_t::RegistrationState

8.524.2.13 uint32_t unpack_nas_GetServingNetwork_t::Roaming

8.525 unpack_nas_GetServingNetworkCapabilities_t Struct Reference

Data Fields

- uint8_t DataCapsLen
- uint8_t DataCaps [255]

8.525.1 Detailed Description

Parameters

DataCapsLen	data capabilities len
DataCap	data capabilities

8.525.2 Field Documentation

8.525.2.1 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCaps[255]

8.525.2.2 uint8_t unpack_nas_GetServingNetworkCapabilities_t::DataCapsLen

8.526 unpack_nas_GetSignalStrengths_t Struct Reference

Data Fields

- uint32_t len
- signed char rssi [8]

- uint32_t radio [8]

8.526.1 Detailed Description

Parameters

len	number of rssi & radio items following
rssi	signal strength array
radio	radio interface array

8.526.2 Field Documentation

8.526.2.1 uint32_t unpack_nas_GetSignalStrengths_t::len

8.526.2.2 uint32_t unpack_nas_GetSignalStrengths_t::radio[8]

8.526.2.3 signed char unpack_nas_GetSignalStrengths_t::rssi[8]

8.527 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

8.527.1 Detailed Description

Parameters

InstanceSize	total instances
Instances	info for instances

8.527.2 Field Documentation

8.527.2.1 nas_QmiNas3GppNetworkInfo* unpack_nas_PerformNetworkScan_t::p3GppNetworkInfoInstances

8.527.2.2 uint8_t* unpack_nas_PerformNetworkScan_t::p3GppNetworkInstanceSize

8.527.2.3 nas_QmisNasPcsDigit* unpack_nas_PerformNetworkScan_t::pPCSInstance

8.527.2.4 uint8_t* unpack_nas_PerformNetworkScan_t::pPCSInstanceSize

8.527.2.5 nas_QmiNas3GppNetworkRAT* unpack_nas_PerformNetworkScan_t::pRATINstance

8.527.2.6 uint8_t* unpack_nas_PerformNetworkScan_t::pRATInstanceSize

8.527.2.7 uint32_t* unpack_nas_PerformNetworkScan_t::pScanResult

8.528 unpack_nas_SetDataCapabilitiesCallback_ind_t Struct Reference

Data Fields

- uint8_t dataCapsSize
- uint8_t dataCaps [255]

8.528.1 Detailed Description

Parameters

dataCapsSize	Number of Data Capabilities
dataCaps	Data Capabilities

8.528.2 Field Documentation

8.528.2.1 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCaps[255]

8.528.2.2 uint8_t unpack_nas_SetDataCapabilitiesCallback_ind_t::dataCapsSize

8.529 unpack_nas_SetEventReportInd_t Struct Reference

Data Fields

- nas_SignalStrengthTlv SSTlv
- nas_RFInfoTlv RFTlv
- nas_RejectReasonTlv RRTlv
- nas_SLQSSignalStrengthsTlv SLQSSSTlv

8.529.1 Detailed Description

Parameters

SSTlv	signal strength tlv
RFTlv	RF tlv
RRTlv	RR tlv
SLQSSSTlv	signal strength complete info tlv

8.529.2 Field Documentation

8.529.2.1 nas_RFInfoTlv unpack_nas_SetEventReportInd_t::RFTlv

8.529.2.2 nas_RejectReasonTlv unpack_nas_SetEventReportInd_t::RRTlv

8.529.2.3 nas_SLQSSignalStrengthsTlv unpack_nas_SetEventReportInd_t::SLQSSSTlv

8.529.2.4 nas_SignalStrengthTlv unpack_nas_SetEventReportInd_t::SSTlv

8.530 unpack_nas_SetNasLTECphyCaIndCallback_ind_t Struct Reference

Data Fields

- nas_PhyCaAggScellIndType sPhyCaAggScellIndType

- nas_PhyCaAggScellDlBw sPhyCaAggScellDlBw
- nas_PhyCaAggScellInfo sPhyCaAggScellInfo
- nas_PhyCaAggPcellInfo sPhyCaAggPcellInfo
- nas_PhyCaAggScellIndex sPhyCaAggScellIndex

8.530.1 Detailed Description

Structure for storing the LTEC PHY CA indication parameters.

Parameters

pPhyCaAggScellIndType	• See nas_PhyCaAggScellIndType for more information.
sPhyCaAggScellDlBw	• See nas_PhyCaAggScellDlBw for more information.
sPhyCaAggScellInfo	• See nas_PhyCaAggScellInfo for more information.
sPhyCaAggPcellInfo	• See nas_PhyCaAggPcellInfo for more information.
sPhyCaAggScellIndex	• See nas_PhyCaAggScellIndex for more information.

8.530.2 Field Documentation

8.530.2.1 nas_PhyCaAggPcellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggPcellInfo

8.530.2.2 nas_PhyCaAggScellDlBw unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellDlBw

8.530.2.3 nas_PhyCaAggScellIndex unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndex

8.530.2.4 nas_PhyCaAggScellIndType unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellIndType

8.530.2.5 nas_PhyCaAggScellInfo unpack_nas_SetNasLTECphyCaIndCallback_ind_t::sPhyCaAggScellInfo

8.531 unpack_nas_SetNetworkPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.531.1 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

8.531.2 Field Documentation

8.531.2.1 uint16_t unpack_nas_SetNetworkPreference_t::Tlvresult

8.532 unpack_nas_SetRoamingIndicatorCallback_ind_t Struct Reference

Data Fields

- uint8_t roaming

8.532.1 Detailed Description

Parameters

roaming	• Roaming Indication – 0 - Roaming – 1 - Home – 2 - Roaming partner – >2 - Operator defined values

8.532.2 Field Documentation

8.532.2.1 uint8_t unpack_nas_SetRoamingIndicatorCallback_ind_t::roaming

8.533 unpack_nas_SetServingSystemCallback_ind_t Struct Reference

Data Fields

- NASServingSystemInfo SSInfo
- uint16_t TlvResult

8.533.1 Detailed Description

Parameters

SSInfo	• Serving system parameters information – See NASServingSystemInfo for more details
TlvResult	• unpack result

8.533.2 Field Documentation

8.533.2.1 NASServingSystemInfo unpack_nas_SetServingSystemCallback_ind_t::SSInfo

8.533.2.2 uint16_t unpack_nas_SetServingSystemCallback_ind_t::TlvResult

8.534 unpack_nas_SLQSGetErrorRate_t Struct Reference

Data Fields

- uint16_t * pCDMAFrameErrRate
- uint16_t * pHDRPackErrRate
- uint8_t * pGSMBER
- uint8_t * pWCDMABER

8.534.1 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.

8.534.2 Field Documentation

8.534.2.1 uint16_t* unpack_nas_SLQSGetErrorRate_t::pCDMAFrameErrRate

8.534.2.2 uint8_t* unpack_nas_SLQSGetErrorRate_t::pGSMBER

8.534.2.3 uint16_t* unpack_nas_SLQSGetErrorRate_t::pHDRPackErrRate

8.534.2.4 uint8_t* unpack_nas_SLQSGetErrorRate_t::pWCDMABER

8.535 unpack_nas_SlqsGetLTECphyCAInfo_t Struct Reference

Data Fields

- NasGetLTECphyCaInfo LTECphyCaInfo
- uint16_t Tlvresult

8.535.1 Detailed Description

Parameters

LTECphyCa	• Carrier aggregation event information – See NasGetLTECphyCaInfo for more details
Tlvresult	• unpack result

8.535.2 Field Documentation

8.535.2.1 NasGetLTECphyCaInfo unpack_nas_SlqsGetLTECphyCAInfo_t::LTECphyCaInfo

8.535.2.2 uint16_t unpack_nas_SlqsGetLTECphyCAInfo_t::Tlvresult

8.536 unpack_nas_SLQSGetNetworkTime_t Struct Reference

Data Fields

- nas_timeInfo * p3GPP2TimeInfo
- nas_timeInfo * p3GPPTimeInfo

8.536.1 Detailed Description

This structure contains information about the GetNetworkTime response parameters.

Parameters

p3GPP2TimeInfo	[Optional] • See nas_timeInfo for more information
p3GPPTimeInfo	[Optional] • See nas_timeInfo for more information

8.536.2 Field Documentation

8.536.2.1 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPP2TimeInfo

8.536.2.2 nas_timeInfo* unpack_nas_SLQSGetNetworkTime_t::p3GPPTimeInfo

8.537 unpack_nas_SLQSGetOperatorNameData_t Struct Reference

Data Fields

- nas_serviceProviderName * pSrvcProviderName
- nas_operatorPLMNList * pOperatorPLMNList
- nas_PLMNNetworkName * pPLMNNetworkName
- nas_operatorNameString * pOperatorNameString
- nas_PLMNNetworkNameData * pNITZInformation

8.537.1 Detailed Description

Structure for Operator Name unpack.

Parameters

pSrvcProviderName	• 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.

pOperatorName-String	• Refer nas_operatorNameString for details (Optional). • Can provide NULL if this parameter is not required.
pNITZ-Information	• Refer nas_PLMNNetworkNameData for details (Optional). • Can provide NULL if this parameter is not required.

8.537.2 Field Documentation

8.537.2.1 nas_PLMNNetworkNameData∗ unpack_nas_SLQSGetOperatorNameData_t::pNITZInformation

8.537.2.2 nas_operatorNameString∗ unpack_nas_SLQSGetOperatorNameData_t::pOperatorNameString

8.537.2.3 nas_operatorPLMNList∗ unpack_nas_SLQSGetOperatorNameData_t::pOperatorPLMNList

8.537.2.4 nas_PLMNNetworkName∗ unpack_nas_SLQSGetOperatorNameData_t::pPLMNNetworkName

8.537.2.5 nas_serviceProviderName∗ unpack_nas_SLQSGetOperatorNameData_t::pSrvcProviderName

8.538 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]

8.538.1 Field Documentation

8.538.1.1 char unpack_nas_SLQSGetPLMNName_t::longName[255]

8.538.1.2 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameCI

8.538.1.3 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameEn

8.538.1.4 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameLen

8.538.1.5 uint8_t unpack_nas_SLQSGetPLMNName_t::longNameSB

8.538.1.6 uint8_t unpack_nas_SLQSGetPLMNName_t::shortName[255]

8.538.1.7 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameCI

8.538.1.8 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameEn

8.538.1.9 char unpack_nas_SLQSGetPLMNName_t::shortNameLen

8.538.1.10 uint8_t unpack_nas_SLQSGetPLMNName_t::shortNameSB

8.538.1.11 char unpack_nas_SLQSGetPLMNName_t::spn[255]

8.538.1.12 uint8_t unpack_nas_SLQSGetPLMNName_t::spnEncoding

8.538.1.13 uint8_t unpack_nas_SLQSGetPLMNName_t::spnLength

8.539 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

8.539.1 Detailed Description

Parameters

ServingSystem	serving system info
RoamIndicator-Val	roaming indicator value
DataSrv-Capabilities	data servcie capabilities
CurrentPLMN	current PLMN info
SystemID	system id

NetworkID	network id
BasestationID	base station id
Basestation-Latitude	base station latitude
Basestation-Longitude	base station longitude
Roaming-IndicatorList	roaming indicator list
DefaultRoamInd	default roaming indicator
3Gpp2TimeZone	3Gpp2 time zone
pCDMA_P_Rev	cdma P_Rev in use
3GppTimeZone	3Gpp time zone
GppNetworkDS-TAdjustment	3GPP network daylight saving adjustment
Lac	location area code
CellID	3GPP cell id
ConcSvcInfo	3GPP2 concurrent servcie info
PRLInd	3GPP2 PRL indicator
DTMInd	DTM indicator(GSM)
DetailedSvcInfo	detail servcie info
CDMASystem-InfoExt	extra cdma system info
HdrPersonality	hdr personality
TrackAreaCode	track area code
CallBarStatus	call barring status

8.539.2 Field Documentation

8.539.2.1 uint16_t unpack_nas_SLQSGetServingSystem_t::BasestationID

8.539.2.2 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLatitude

8.539.2.3 uint32_t unpack_nas_SLQSGetServingSystem_t::BasestationLongitude

8.539.2.4 nas_callBarStatus unpack_nas_SLQSGetServingSystem_t::CallBarStatus

8.539.2.5 uint8_t unpack_nas_SLQSGetServingSystem_t::CDMA_P_Rev

8.539.2.6 nas_CDMASysInfoExt unpack_nas_SLQSGetServingSystem_t::CDMASystemInfoExt

8.539.2.7 uint32_t unpack_nas_SLQSGetServingSystem_t::CellID

8.539.2.8 uint8_t unpack_nas_SLQSGetServingSystem_t::ConcSvcInfo

8.539.2.9 nas_currentPLMN unpack_nas_SLQSGetServingSystem_t::CurrentPLMN

8.539.2.10 nas_dataSrvCapabilities unpack_nas_SLQSGetServingSystem_t::DataSrvCapabilities

8.539.2.11 uint8_t unpack_nas_SLQSGetServingSystem_t::DefaultRoamInd

8.539.2.12 nas_detailSvcInfo unpack_nas_SLQSGetServingSystem_t::DetailedSvcInfo

8.539.2.13 uint8_t unpack_nas_SLQSGetServingSystem_t::DTMInd

8.539.2.14 nas_qaQmi3Gpp2TimeZone unpack_nas_SLQSGetServingSystem_t::Gpp2TimeZone

8.539.2.15 uint8_t unpack_nas_SLQSGetServingSystem_t::GppNetworkDSTAdjustment

8.539.2.16 uint8_t unpack_nas_SLQSGetServingSystem_t::GppTimeZone

8.539.2.17 uint8_t unpack_nas_SLQSGetServingSystem_t::HdrPersonality

8.539.2.18 uint16_t unpack_nas_SLQSGetServingSystem_t::Lac

8.539.2.19 uint16_t unpack_nas_SLQSGetServingSystem_t::NetworkID

8.539.2.20 uint8_t unpack_nas_SLQSGetServingSystem_t::PRLInd

8.539.2.21 uint8_t unpack_nas_SLQSGetServingSystem_t::RoamIndicatorVal

8.539.2.22 nas_roamIndList unpack_nas_SLQSGetServingSystem_t::RoamingIndicatorList

8.539.2.23 nas_servSystem unpack_nas_SLQSGetServingSystem_t::ServingSystem

8.539.2.24 uint16_t unpack_nas_SLQSGetServingSystem_t::SystemID

8.539.2.25 uint16_t unpack_nas_SLQSGetServingSystem_t::TrackAreaCode

8.540 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

8.540.1 Detailed Description

Parameters

rxSignal-StrengthListLen	number of elements in Receive Signal Strength List
rxSignal-StrengthList	signal strength list
ecioListLen	number of elements in ECIO List
ecioList	ecio list
Io	received Io in dBm; IO is only applicable for 1xEV-DO
sinr	SINR level; SINR is only applicable for 1xEV-DO
errorRateListLen	number of elements in Error Rate List
errorRateList	error rate list

rsrqInfo	rsrq info
ltesnr	lte snr info
ltersrp	lte srp info

8.540.2 Field Documentation

8.540.2.1 nas_ecioListElement unpack_nas_SLQSGetSignalStrength_t::ecioList[18]

8.540.2.2 uint16_t unpack_nas_SLQSGetSignalStrength_t::ecioListLen

8.540.2.3 nas_errorRateListElement unpack_nas_SLQSGetSignalStrength_t::errorRateList[18]

8.540.2.4 uint16_t unpack_nas_SLQSGetSignalStrength_t::errorRateListLen

8.540.2.5 int32_t unpack_nas_SLQSGetSignalStrength_t::lo

8.540.2.6 int16_t unpack_nas_SLQSGetSignalStrength_t::ltersrp

8.540.2.7 int16_t unpack_nas_SLQSGetSignalStrength_t::ltesnr

8.540.2.8 nas_rsrqInformation unpack_nas_SLQSGetSignalStrength_t::rsrqInfo

8.540.2.9 nas_rxSignalStrengthListElement unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthList[18]

8.540.2.10 uint16_t unpack_nas_SLQSGetSignalStrength_t::rxSignalStrengthListLen

8.540.2.11 uint16_t unpack_nas_SLQSGetSignalStrength_t::signalStrengthReqMask

8.540.2.12 uint8_t unpack_nas_SLQSGetSignalStrength_t::sinr

8.541 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

8.541.1 Detailed Description

Parameters

pCDMASrvStatusInfo	• See SrvStatusInfo for more information.
pHDRSrvStatusInfo	• See SrvStatusInfo for more information.
pGSMSrvStatusInfo	• See GSMSrvStatusInfo for more information.
pWCDMASrvStatusInfo	• See GSMSrvStatusInfo for more information.
pLTESrvStatusInfo	• See GSMSrvStatusInfo for more information.
pCDMASysInfo	• See CDMASysInfo for more information.
pHDRSysInfo	• See HDRSysInfo for more information.
pGSMSysInfo	• See GSMSysInfo for more information.
pWCDMASysInfo	• See WCDMASysInfo for more information.
pLTESysInfo	• See LTESysInfo for more information.
pAddCDMASysInfo	• See 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 AddSysInfo for more information.
pAddWCDMASysInfo	• See 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 CallBarringSysInfo for more information.

pWCDMACall-BarringSysInfo	• See 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

8.541.2 Field Documentation

8.541.2.1 nas_AddCDMASysInfo∗ unpack_nas_SLQSGetSysInfo_t::pAddCDMASysInfo

8.541.2.2 nas_AddSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pAddGSMSysInfo

8.541.2.3 uint16_t∗ unpack_nas_SLQSGetSysInfo_t::pAddHDRSysInfo

8.541.2.4 uint16_t∗ unpack_nas_SLQSGetSysInfo_t::pAddLTESysInfo

8.541.2.5 nas_AddSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pAddWCDMASysInfo

8.541.2.6 nas_SrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pCDMASrvStatusInfo

8.541.2.7 nas_CDMASysInfo∗ unpack_nas_SLQSGetSysInfo_t::pCDMASysInfo

8.541.2.8 nas_CallBarringSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pGSMCallBarringSysInfo

8.541.2.9 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pGSMCipherDomainSysInfo

8.541.2.10 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pGSMSrvStatusInfo

8.541.2.11 nas_GSMSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pGSMSysInfo

8.541.2.12 nas_SrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pHDRSrvStatusInfo

8.541.2.13 nas_HDRSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pHDRSysInfo

8.541.2.14 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pLTESrvStatusInfo

8.541.2.15 nas_LTESysInfo∗ unpack_nas_SLQSGetSysInfo_t::pLTESysInfo

8.541.2.16 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pLTEVoiceSupportSysInfo

8.541.2.17 nas_CallBarringSysInfo∗ unpack_nas_SLQSGetSysInfo_t::pWCDMACallBarringSysInfo

8.541.2.18 uint8_t∗ unpack_nas_SLQSGetSysInfo_t::pWCDMACipherDomainSysInfo

8.541.2.19 nas_GSMSrvStatusInfo∗ unpack_nas_SLQSGetSysInfo_t::pWCDMASrvStatusInfo

8.541.2.20 nas_WCDMASysInfo∗ unpack_nas_SLQSGetSysInfo_t::pWCDMASysInfo

8.542 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

8.542.1 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

8.542.2 Field Documentation

8.542.2.1 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pBandPref

8.542.2.2 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pEmerMode

8.542.2.3 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pGWAcqOrderPref

8.542.2.4 uint64_t* unpack_nas_SLQSGetSysSelectionPref_t::pLTEBandPref

8.542.2.5 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pModePref

8.542.2.6 uint8_t* unpack_nas_SLQSGetSysSelectionPref_t::pNetSelPref

8.542.2.7 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pPRLPref

8.542.2.8 uint16_t* unpack_nas_SLQSGetSysSelectionPref_t::pRoamPref

8.542.2.9 uint32_t* unpack_nas_SLQSGetSysSelectionPref_t::pSrvDomainPref

8.543 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

8.543.1 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

8.543.2 Field Documentation

8.543.2.1 nas_CDMAChannel* unpack_nas_SLQSNasGet3GPP2Subscription_t::pCDMAChannel

8.543.2.2 nas_dirNum* unpack_nas_SLQSNasGet3GPP2Subscription_t::pDirNum

8.543.2.3 nas_homeSIDNID* unpack_nas_SLQSNasGet3GPP2Subscription_t::pHomeSIDNID

8.543.2.4 nas_minBasedIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pMinBasedIMSI

8.543.2.5 nas_namName* unpack_nas_SLQSNasGet3GPP2Subscription_t::pNAMNameInfo

8.543.2.6 nas_trueIMSI* unpack_nas_SLQSNasGet3GPP2Subscription_t::pTrueIMSI

8.544 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

8.544.1 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-NeighboringGS-M	• See nas_LTEInfoNeighboringGSM for more information.
pLTEInfo-NeighboringWC-DMA	• See nas_LTEInfoNeighboringWCDMA for more information.
pUMTSCellID	• Cell ID. • 0xFFFFFFFF indicates cell ID information is not present.
pWCDMAInfoLT-ENeighborCell	• See nas_WCDMAInfoLTENeighborCell for more information.

8.544.2 Field Documentation

8.544.2.1 nas_CDMAInfo∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pCDMAInfo

8.544.2.2 nas_GERANInfo∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pGERANInfo

8.544.2.3 nas_LTEInfoInterfreq∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoInterfreq

8.544.2.4 nas_LTEInfoIntrafreq∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoIntrafreq

8.544.2.5 nas_LTEInfoNeighboringGSM∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringGSM

8.544.2.6 nas_LTEInfoNeighboringWCDMA∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pLTEInfoNeighboringWCDMA

8.544.2.7 uint32_t∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSCellID

8.544.2.8 nas_UMTSInfo∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pUMTSInfo

8.544.2.9 nas_WCDMAInfoLTENeighborCell∗ unpack_nas_SLQSNasGetCellLocationInfo_t::pWCDMAInfoLTENeighbor-Cell

8.545 unpack_nas_SLQSNasGetHDRColorCode_t Struct Reference

Data Fields

- uint8_t * pColorCode

8.545.1 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

8.545.2 Field Documentation

8.545.2.1 uint8_t* unpack_nas_SLQSNasGetHDRColorCode_t::pColorCode

8.546 unpack_nas_SLQSNasGetSigInfo_t Struct Reference

Data Fields

- cdmaSSInfo CDMASSInfo
- hdrSSInfo HDRSSInfo
- int8_t GSMSSInfo
- cdmaSSInfo WCDMASSInfo
- lteSSInfo LTESSInfo

8.546.1 Detailed Description

Parameters

CDMASSInfo	CDMA Signal Strength Information
HDRSSInfo	HDR Signal Strength Information
GSMSSInfo	GSM signal strength is the RSSI in dBm.
WCDMASSInfo	WCDMA Signal Strength Information
LTESSInfo	LTE Signal Strength Information

8.546.2 Field Documentation

8.546.2.1 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::CDMASSInfo

8.546.2.2 int8_t unpack_nas_SLQSNasGetSigInfo_t::GSMSSInfo

8.546.2.3 hdrSSInfo unpack_nas_SLQSNasGetSigInfo_t::HDRSSInfo

8.546.2.4 lteSSInfo unpack_nas_SLQSNasGetSigInfo_t::LTESSInfo

8.546.2.5 cdmaSSInfo unpack_nas_SLQSNasGetSigInfo_t::WCDMASSInfo

8.547 unpack_nas_SLQSNasGetTxRxInfo_t Struct Reference

Data Fields

- nas_rxInfo * pRXChain0Info
- nas_rxInfo * pRXChain1Info
- nas_txInfo * pTXInfo

8.547.1 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.

8.547.2 Field Documentation

8.547.2.1 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain0Info

8.547.2.2 nas_rxInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pRXChain1Info

8.547.2.3 nas_txInfo* unpack_nas_SLQSNasGetTxRxInfo_t::pTXInfo

8.548 unpack_nas_SLQSNasNetworkTimeCallBack_ind_t Struct Reference

Data Fields

- nas_UniversalTime universalTime
- uint8_t * pTimeZone
- uint8_t * pDayltSavAdj
- uint8_t * pRadioInterface

8.548.1 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

8.548.2 Field Documentation

8.548.2.1 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pDayltSavAdj

8.548.2.2 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pRadioInterface

8.548.2.3 uint8_t* unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::pTimeZone

8.548.2.4 nas_UniversalTime unpack_nas_SLQSNasNetworkTimeCallBack_ind_t::universalTime

8.549 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

8.549.1 Detailed Description

Parameters

pCDMASigInfo	CDMA SS info
pHDRSigInfo	HDR SS info
pGSMSigInfo	GSM signal info
pWCDMASigInfo	WCDMA signal info
pLTESigInfo	LTE signal info

pRscp	RSCP of the Primary Common Control Physical Channel
pTDSCDMASig-InfoExt	extra CDMA sig info

8.549.2 Field Documentation

8.549.2.1 cdmaSSInfo∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pCDMASigInfo

8.549.2.2 int8_t∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pGSMSigInfo

8.549.2.3 hdrSSInfo∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pHDRSigInfo

8.549.2.4 lteSSInfo∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pLTESigInfo

8.549.2.5 int8_t∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pRscp

8.549.2.6 tdscdmaSigInfoExt∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pTDSCDMASigInfoExt

8.549.2.7 cdmaSSInfo∗ unpack_nas_SLQSNasSigInfoCallback_ind_t::pWCDMASigInfo

8.550 unpack_nas_SLQSNASSwiGetChannelLock_t Struct Reference

Data Fields

- nas_wcdmaUARFCN ∗ pWcdmaUARFCN
- nas_lteEARFCN ∗ pLteEARFCN
- nas_ltePCI ∗ pLtePCI

8.550.1 Detailed Description

Structure for Get Channel Lock unpack.

Parameters

pWcdmaUARF-CN	[Optional] • See nas_wcdmaUARFCN for more information
pLteEARFCN	[Optional] • See nas_lteEARFCN for more information
pLtePCI	[Optional] • See nas_ltePCI for more information

8.550.2 Field Documentation

8.550.2.1 nas_lteEARFCN∗ unpack_nas_SLQSNASSwiGetChannelLock_t::pLteEARFCN

8.550.2.2 nas_ltePCI∗ unpack_nas_SLQSNASSwiGetChannelLock_t::pLtePCI

8.550.2.3 nas_wcdmaUARFCN∗ unpack_nas_SLQSNASSwiGetChannelLock_t::pWcdmaUARFCN

8.551 unpack_nas_SLQSNasSwiModemStatus_t Struct Reference

Data Fields

- nas_CommInfo commonInfo
- nas_LTEInfo * pLTEInfo

8.551.1 Detailed Description

Structure for storing the SLQS Nas Swi Modem Status response parameters.

Parameters

commonInfo	(mandatory) • See CommInfo for more information
pLTEInfo	(optional) • See LTEInfo for more information

8.551.2 Field Documentation

8.551.2.1 nas_CommInfo unpack_nas_SLQSNasSwiModemStatus_t::commonInfo

8.551.2.2 nas_LTEInfo* unpack_nas_SLQSNasSwiModemStatus_t::pLTEInfo

8.552 unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSwiOTAMessageInd Info
- uint16_t Tlvresult

8.552.1 Detailed Description

Parameters

Info	• Structure used to store all QMI Notification Info. – See NASQmiCbkNasSwiOTAMessageInd for more details
Tlvresult	• unpack result

8.552.2 Field Documentation

8.552.2.1 NASQmiCbkNasSwiOTAMessageInd unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Info

8.552.2.2 uint16_t unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t::Tlvresult

8.553 unpack_nas_SLQSNasTimerCallback_ind_t Struct Reference

Data Fields

- char t3396_apn [101]
- uint8_t t3396_plmn_id [3]
- uint32_t t3396_val

8.553.1 Detailed Description

Structure for Network Timer indication parameters.

Parameters

t3396_apn	• apn
t3396_plmn_id	• plmn id
t3396_val	• timer value

8.553.2 Field Documentation

8.553.2.1 char unpack_nas_SLQSNasTimerCallback_ind_t::t3396_apn[101]

8.553.2.2 uint8_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_plmn_id[3]

8.553.2.3 uint32_t unpack_nas_SLQSNasTimerCallback_ind_t::t3396_val

8.554 unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t Struct Reference

Data Fields

- NASQmiCbkNasSystemSelPrefInd Info
- uint16_t Tlvresult

8.554.1 Detailed Description

Parameters

Info	• Structure used to store all QMI Notification Info. – See NASQmiCbkNasSystemSelPrefInd for more details
Tlvresult	• unpack result

8.554.2 Field Documentation

8.554.2.1 NASQmiCbkNasSystemSelPrefInd unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Info

8.554.2.2 uint16_t unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t::Tlvresult

8.555 unpack_nas_SLQSSwiGetHDRPersonality_t Struct Reference

Data Fields

- uint16_t * pCurrentPersonality
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtocolSubtypeElement

8.555.1 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.

8.555.2 Field Documentation

8.555.2.1 uint16_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pCurrentPersonality

8.555.2.2 uint8_t* unpack_nas_SLQSSwiGetHDRPersonality_t::pPersonalityListLength

8.555.2.3 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRPersonality_t::pProtocolSubtypeElement

8.556 unpack_nas_SLQSSwiGetHDRProtSubtype_t Struct Reference

Data Fields

- uint16_t * pCurrentPrsnlty
- uint8_t * pPersonalityListLength
- nas_protocolSubtypeElement * pProtoSubTypElmnt
- uint64_t * pAppSubType

8.556.1 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,

8.556.2 Field Documentation

8.556.2.1 uint64_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pAppSubType

8.556.2.2 uint16_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pCurrentPrsnlty

8.556.2.3 uint8_t* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pPersonalityListLength

8.556.2.4 nas_protocolSubtypeElement* unpack_nas_SLQSSwiGetHDRProtSubtype_t::pProtoSubTypElmnt

8.557 unpack_nas_SLQSSwiGetHRPDStats_t Struct Reference

Data Fields

- nas_DRCParams * pDRCParams
- uint8_t * pUATI
- nas_PilotSetData * pPilotSetData

8.557.1 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.

8.557.2 Field Documentation

8.557.2.1 nas_DRCParams∗ unpack_nas_SLQSSwiGetHRPDStats_t::pDRCParams

8.557.2.2 nas_PilotSetData∗ unpack_nas_SLQSSwiGetHRPDStats_t::pPilotSetData

8.557.2.3 uint8_t∗ unpack_nas_SLQSSwiGetHRPDStats_t::pUATI

8.558 unpack_nas_SLQSSwiGetLteCQI_t Struct Reference

Data Fields

- uint8_t ValidityCW0
- uint8_t CQIValueCW0
- uint8_t ValidityCW1
- uint8_t CQIValueCW1

8.558.1 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

8.558.2 Field Documentation

8.558.2.1 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW0

8.558.2.2 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::CQIValueCW1

8.558.2.3 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW0

8.558.2.4 uint8_t unpack_nas_SLQSSwiGetLteCQI_t::ValidityCW1

8.559 unpack_nas_SLQSSwiGetLteSccRxInfo_t Struct Reference

Data Fields

- nas_SccRxInfo ∗ pSccRxInfo

8.559.1 Detailed Description

Parameters

pSccRxInfo	Secondary carrier Rx signal level info

8.559.2 Field Documentation

8.559.2.1 nas_SccRxInfo∗ unpack_nas_SLQSSwiGetLteSccRxInfo_t::pSccRxInfo

8.560 unpack_nas_SLQSSwiNetworkDebug_t Struct Reference

Data Fields

- uint8_t ∗ pObjectVer
- nas_NetworkStat1x ∗ pNetworkStat1x
- nas_NetworkStatEVDO ∗ pNetworkStatEVDO
- nas_DeviceConfigDetail ∗ pDeviceConfigDetail
- nas_DataStatusDetail ∗ pDataStatusDetail

8.560.1 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
pNetworkStatEVDO	• See nas_NetworkStatEVDO for more information.
pDeviceConfigDetail	• See nas_DeviceConfigDetail for more information.
pDataStatusDetail	• See nas_DataStatusDetail for more information.

8.560.2 Field Documentation

8.560.2.1 nas_DataStatusDetail∗ unpack_nas_SLQSSwiNetworkDebug_t::pDataStatusDetail

8.560.2.2 nas_DeviceConfigDetail∗ unpack_nas_SLQSSwiNetworkDebug_t::pDeviceConfigDetail

8.560.2.3 nas_NetworkStat1x∗ unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStat1x

8.560.2.4 nas_NetworkStatEVDO∗ unpack_nas_SLQSSwiNetworkDebug_t::pNetworkStatEVDO

8.560.2.5 uint8_t* unpack_nas_SLQSSwiNetworkDebug_t::pObjectVer

8.561 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

8.561.1 Detailed Description

Parameters

pCDMASrvStatusInfo	• See SrvStatusInfo for more information.
pHDRSrvStatusInfo	• See SrvStatusInfo for more information.
pGSMSrvStatusInfo	• See GSMSrvStatusInfo for more information.
pWCDMASrvStatusInfo	• See GSMSrvStatusInfo for more information.
pLTESrvStatusInfo	• See GSMSrvStatusInfo for more information.
pCDMASysInfo	• See CDMASysInfo for more information.
pHDRSysInfo	• See HDRSysInfo for more information.

pGSMSysInfo	• See GSMSysInfo for more information.
pWCDMASysInfo	• See WCDMASysInfo for more information.
pLTESysInfo	• See LTESysInfo for more information.
pAddCDMASysInfo	• See 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 AddSysInfo for more information.
pAddWCDMASysInfo	• See 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 CallBarringSysInfo for more information.
pWCDMACallBarringSysInfo	• See 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

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

8.561.2 Field Documentation

8.561.2.1 **nas_AddCDMASysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddCDMASysInfo

8.561.2.2 **nas_AddSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddGSMSysInfo

8.561.2.3 uint16_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddHDRSysInfo

8.561.2.4 uint16_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddLTESysInfo

8.561.2.5 **nas_AddSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pAddWCDMASysInfo

8.561.2.6 **nas_SrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASrvStatusInfo

8.561.2.7 **nas_CDMASysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pCDMASysInfo

8.561.2.8 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCallBarringSysInfo

8.561.2.9 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMCipherDomainSysInfo

8.561.2.10 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSrvStatusInfo

8.561.2.11 **nas_GSMSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pGSMSysInfo

8.561.2.12 **nas_SrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSrvStatusInfo

8.561.2.13 **nas_HDRSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pHDRSysInfo

8.561.2.14 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTESrvStatusInfo

8.561.2.15 **nas_LTESysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTESysInfo

8.561.2.16 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pLTEVoiceSupportSysInfo

8.561.2.17 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pSysInfoNoChange

8.561.2.18 **nas_CallBarringSysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACallBarringSysInfo

8.561.2.19 uint8_t∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMACipherDomainSysInfo

8.561.2.20 **nas_GSMSrvStatusInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASrvStatusInfo

8.561.2.21 **nas_WCDMASysInfo**∗ unpack_nas_SLQSSysInfoCallback_ind_t::pWCDMASysInfo

8.562 unpack_omaDmConfigTlv_t Struct Reference

Data Fields

- uint8_t state
- uint8_t userInputReq
- uint16_t userInputTimeout
- uint16_t alertmsglength
- uint8_t alertmsg [256]

8.562.1 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
user_input_req	- Bit mask of available user inputs • 0x00 - No user input required.Informational indication • 0x01 - Accept • 0x02 - Reject
user_input_-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)

8.562.2 Field Documentation

8.562.2.1 uint8_t unpack_omaDmConfigTlv_t::alertmsg[256]

8.562.2.2 uint16_t unpack_omaDmConfigTlv_t::alertmsglength

8.562.2.3 uint8_t unpack_omaDmConfigTlv_t::state

8.562.2.4 uint8_t unpack_omaDmConfigTlv_t::userInputReq

8.562.2.5 uint16_t unpack_omaDmConfigTlv_t::userInputTimeout

8.563 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

8.563.1 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
user_input_req	- Bit mask of available user inputs • 0x00 - No user input required.Informational indication • 0x01 - Accept • 0x02 - Reject
user_input_timeout	• Timeout for user input in minutes. A value of 0 means no time-out
fw_dload_size	• The size (in bytes) of the firmware update package
fw_dload_complete	• The number of bytes downloaded. Need to determine how often to send this message for progress bar notification. Every 500ms or 5% increment.
update_complete_status	• See table below.
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

8.563.2 Field Documentation

8.563.2.1 uint8_t unpack_omaDmFotaTlv_t::description[256]

8.563.2.2 uint16_t unpack_omaDmFotaTlv_t::descriptionlength

8.563.2.3 uint32_t unpack_omaDmFotaTlv_t::fwdloadsize

8.563.2.4 uint32_t unpack_omaDmFotaTlv_t::fwloadComplete

8.563.2.5 uint16_t unpack_omaDmFotaTlv_t::namelength

8.563.2.6 uint8_t unpack_omaDmFotaTlv_t::package_name[256]

8.563.2.7 uint8_t unpack_omaDmFotaTlv_t::sessionType

8.563.2.8 uint8_t unpack_omaDmFotaTlv_t::severity

8.563.2.9 uint8_t unpack_omaDmFotaTlv_t::state

8.563.2.10 uint16_t unpack_omaDmFotaTlv_t::updateCompleteStatus

8.563.2.11 uint8_t unpack_omaDmFotaTlv_t::userInputReq

8.563.2.12 uint16_t unpack_omaDmFotaTlv_t::userInputTimeout

8.563.2.13 uint8_t unpack_omaDmFotaTlv_t::version[256]

8.563.2.14 uint16_t unpack_omaDmFotaTlv_t::versionlength

8.564 unpack_omaDmNotificationsTlv_t Struct Reference

Data Fields

- uint8_t notification
- uint16_t sessionStatus

8.564.1 Field Documentation

8.564.1.1 uint8_t unpack_omaDmNotificationsTlv_t::notification

8.564.1.2 uint16_t unpack_omaDmNotificationsTlv_t::sessionStatus

8.565 unpack_pds_GetPDSDefaults_t Struct Reference

Data Fields

- uint32_t * pOperation
- uint8_t * pTimeout
- uint32_t * pInterval
- uint32_t * pAccuracy

8.565.1 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)

8.565.2 Field Documentation

8.565.2.1 uint32_t* unpack_pds_GetPDSDefaults_t::pAccuracy

8.565.2.2 uint32_t* unpack_pds_GetPDSDefaults_t::pInterval

8.565.2.3 uint32_t* unpack_pds_GetPDSDefaults_t::pOperation

8.565.2.4 uint8_t* unpack_pds_GetPDSDefaults_t::pTimeout

8.566 unpack_pds_GetPDSState_t Struct Reference

Data Fields

- uint32_t * pEnabledStatus
- uint32_t * pTrackingStatus

8.566.1 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

8.566.2 Field Documentation

8.566.2.1 uint32_t* unpack_pds_GetPDSState_t::pEnabledStatus

8.566.2.2 uint32_t* unpack_pds_GetPDSState_t::pTrackingStatus

8.567 unpack_pds_GetPortAutomaticTracking_t Struct Reference

Data Fields

- uint32_t * pbAuto

8.567.1 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

8.567.2 Field Documentation

8.567.2.1 uint32_t* unpack_pds_GetPortAutomaticTracking_t::pbAuto

8.568 unpack_qmi_t Struct Reference

Data Fields

- enum msgtype type

- uint16_t msgid
- uint16_t xid

8.568.1 Detailed Description

qmi response context

Parameters

out	*type*	message type
out	*msgid*	message id
out	*xid*	transaction id

8.568.2 Field Documentation

8.568.2.1 uint16_t unpack_qmi_t::msgid

8.568.2.2 enum msgtype unpack_qmi_t::type

8.568.2.3 uint16_t unpack_qmi_t::xid

8.569 unpack_qos_dataRate_t Struct Reference

Data Fields

- uint32_t dataRateMax
- uint32_t guaranteedRate

8.569.1 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)

8.569.2 Field Documentation

8.569.2.1 uint32_t unpack_qos_dataRate_t::dataRateMax

8.569.2.2 uint32_t unpack_qos_dataRate_t::guaranteedRate

8.570 unpack_qos_IPv4Addr_t Struct Reference

Data Fields

- uint32_t addr
- uint32_t subnetMask

8.570.1 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

8.570.2 Field Documentation

8.570.2.1 uint32_t unpack_qos_IPv4Addr_t::addr

8.570.2.2 uint32_t unpack_qos_IPv4Addr_t::subnetMask

8.571 unpack_qos_IPv6Addr_t Struct Reference

Data Fields

- uint8_t addr [16]
- uint8_t prefixLen

8.571.1 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.

8.571.2 Field Documentation

8.571.2.1 uint8_t unpack_qos_IPv6Addr_t::addr[16]

8.571.2.2 uint8_t unpack_qos_IPv6Addr_t::prefixLen

8.572 unpack_qos_IPv6TrafCls_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.572.1 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

8.572.2 Field Documentation

8.572.2.1 uint8_t unpack_qos_IPv6TrafCls_t::mask

8.572.2.2 uint8_t unpack_qos_IPv6TrafCls_t::val

8.573 unpack_qos_pktErrRate_t Struct Reference

Data Fields

- uint16_t multiplier
- uint16_t exponent

8.573.1 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)

8.573.2 Field Documentation

8.573.2.1 uint16_t unpack_qos_pktErrRate_t::exponent

8.573.2.2 uint16_t unpack_qos_pktErrRate_t::multiplier

8.574 unpack_qos_Port_t Struct Reference

Data Fields

- uint16_t port
- uint16_t range

8.574.1 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

8.574.2 Field Documentation

8.574.2.1 uint16_t unpack_qos_Port_t::port

8.574.2.2 uint16_t unpack_qos_Port_t::range

8.575 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

8.575.1 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.

8.575.2 Field Documentation

8.575.2.1 uint8_t unpack_qos_QosFlowInfo_t::BearerID

8.575.2.2 uint8_t unpack_qos_QosFlowInfo_t::is_RxQFlowGranted_Available

8.575.2.3 uint8_t unpack_qos_QosFlowInfo_t::is_TxQFlowGranted_Available

8.575.2.4 uint8_t unpack_qos_QosFlowInfo_t::NumRxFilters

8.575.2.5 uint8_t unpack_qos_QosFlowInfo_t::NumTxFilters

8.575.2.6 unpack_qos_QosFlowInfoState_t unpack_qos_QosFlowInfo_t::QFlowState

8.575.2.7 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::RxQFilter[25]

8.575.2.8 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::RxQFlowGranted

8.575.2.9 unpack_qos_swiQosFilter_t unpack_qos_QosFlowInfo_t::TxQFilter[25]

8.575.2.10 unpack_qos_swiQosFlow_t unpack_qos_QosFlowInfo_t::TxQFlowGranted

8.576 unpack_qos_QosFlowInfoState_t Struct Reference

Data Fields

- uint32_t id
- uint8_t isNewFlow
- uint8_t state

8.576.1 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

8.576.2 Field Documentation

8.576.2.1 uint32_t unpack_qos_QosFlowInfoState_t::id

8.576.2.2 uint8_t unpack_qos_QosFlowInfoState_t::isNewFlow

8.576.2.3 uint8_t unpack_qos_QosFlowInfoState_t::state

8.577 unpack_qos_SLQSQosGetNetworkStatus_t Struct Reference

Data Fields

- uint8_t NWQoSStatus

8.577.1 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

8.577.2 Field Documentation

8.577.2.1 uint8_t unpack_qos_SLQSQosGetNetworkStatus_t::NWQoSStatus

8.578 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

8.578.1 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

8.578.2 Field Documentation

8.578.2.1 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl

8.578.2.2 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext

8.578.2.3 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_dl_ext2

8.578.2.4 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul

8.578.2.5 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext

8.578.2.6 uint8_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::ambr_ul_ext2

8.578.2.7 uint32_t unpack_qos_SLQSQosSwiReadApnExtraParams_t::apnId

8.579 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]

8.579.1 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

8.579.2 Field Documentation

8.579.2.1 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::apnId

8.579.2.2 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::numQosFlow

8.579.2.3 unpack_QosFlowStat_t unpack_qos_SLQSQosSwiReadDataStats_t::qosFlow[10]

8.579.2.4 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_bytes

8.579.2.5 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_rx_pkt

8.579.2.6 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes

8.579.2.7 uint64_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_bytes_drp

8.579.2.8 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt

8.579.2.9 uint32_t unpack_qos_SLQSQosSwiReadDataStats_t::total_tx_pkt_drp

8.580 unpack_qos_SLQSSetQosEventCallback_ind_t Struct Reference

Data Fields

- uint8_t NumFlows
- unpack_qos_QosFlowInfo_t QosFlowInfo [8]

8.580.1 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

8.580.2 Field Documentation

8.580.2.1 uint8_t unpack_qos_SLQSSetQosEventCallback_ind_t::NumFlows

8.580.2.2 unpack_qos_QosFlowInfo_t unpack_qos_SLQSSetQosEventCallback_ind_t::QosFlowInfo[8]

8.581 unpack_qos_SLQSSetQosNWStatusCallback_ind_t Struct Reference

Data Fields

- uint8_t status

8.581.1 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

8.581.2 Field Documentation

8.581.2.1 uint8_t unpack_qos_SLQSSetQosNWStatusCallback_ind_t::status

8.582 unpack_qos_SLQSSetQosPriEventCallback_ind_t Struct Reference

Data Fields

- uint16_t event

8.582.1 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

8.582.2 Field Documentation

8.582.2.1 uint16_t unpack_qos_SLQSSetQosPriEventCallback_ind_t::event

8.583 unpack_qos_SLQSSetQosStatusCallback_ind_t Struct Reference

Data Fields

- uint32_t id
- uint8_t status
- uint8_t event
- uint8_t reason

8.583.1 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

8.583.2 Field Documentation

8.583.2.1 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::event

8.583.2.2 uint32_t unpack_qos_SLQSSetQosStatusCallback_ind_t::id

8.583.2.3 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::reason

8.583.2.4 uint8_t unpack_qos_SLQSSetQosStatusCallback_ind_t::status

8.584 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

8.584.1 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

8.584.2 Field Documentation

8.584.2.1 uint32_t unpack_qos_swiQosFilter_t::EspSpi

8.584.2.2 uint16_t unpack_qos_swiQosFilter_t::Id

8.584.2.3 uint8_t unpack_qos_swiQosFilter_t::index

8.584.2.4 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4DstAddr

8.584.2.5 unpack_qos_IPv4Addr_t unpack_qos_swiQosFilter_t::IPv4SrcAddr

8.584.2.6 unpack_qos_Tos_t unpack_qos_swiQosFilter_t::IPv4Tos

8.584.2.7 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6DstAddr

8.584.2.8 uint32_t unpack_qos_swiQosFilter_t::IPv6Label

8.584.2.9 unpack_qos_IPv6Addr_t unpack_qos_swiQosFilter_t::IPv6SrcAddr

8.584.2.10 unpack_qos_IPv6TrafCls_t unpack_qos_swiQosFilter_t::IPv6TrafCls

8.584.2.11 uint8_t unpack_qos_swiQosFilter_t::is_EspSpi_Available

8.584.2.12 uint8_t unpack_qos_swiQosFilter_t::is_Id_Available

8.584.2.13 uint8_t unpack_qos_swiQosFilter_t::is_IPv4DstAddr_Available

8.584.2.14 uint8_t unpack_qos_swiQosFilter_t::is_IPv4SrcAddr_Available

8.584.2.15 uint8_t unpack_qos_swiQosFilter_t::is_IPv4Tos_Available

8.584.2.16 uint8_t unpack_qos_swiQosFilter_t::is_IPv6DstAddr_Available

8.584.2.17 uint8_t unpack_qos_swiQosFilter_t::is_IPv6Label_Available

8.584.2.18 uint8_t unpack_qos_swiQosFilter_t::is_IPv6SrcAddr_Available

8.584.2.19 uint8_t unpack_qos_swiQosFilter_t::is_IPv6TrafCls_Available

8.584.2.20 uint8_t unpack_qos_swiQosFilter_t::is_NxtHdrProto_Available

8.584.2.21 uint8_t unpack_qos_swiQosFilter_t::is_Precedence_Available

8.584.2.22 uint8_t unpack_qos_swiQosFilter_t::is_TCPDstPort_Available

8.584.2.23 uint8_t unpack_qos_swiQosFilter_t::is_TCPSrcPort_Available

8.584.2.24 uint8_t unpack_qos_swiQosFilter_t::is_TranDstPort_Available

8.584.2.25 uint8_t unpack_qos_swiQosFilter_t::is_TranSrcPort_Available

8.584.2.26 uint8_t unpack_qos_swiQosFilter_t::is_UDPDstPort_Available

8.584.2.27 uint8_t unpack_qos_swiQosFilter_t::is_UDPSrcPort_Available

8.584.2.28 uint8_t unpack_qos_swiQosFilter_t::NxtHdrProto

8.584.2.29 uint16_t unpack_qos_swiQosFilter_t::Precedence

8.584.2.30 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPDstPort

8.584.2.31 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TCPSrcPort

8.584.2.32 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranDstPort

8.584.2.33 unpack_qos_Port_t unpack_qos_swiQosFilter_t::TranSrcPort

8.584.2.34 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPDstPort

8.584.2.35 unpack_qos_Port_t unpack_qos_swiQosFilter_t::UDPSrcPort

8.584.2.36 uint8_t unpack_qos_swiQosFilter_t::version

8.585 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

8.585.1 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

8.585.2 Field Documentation

8.585.2.1 unpack_qos_dataRate_t unpack_qos_swiQosFlow_t::DataRate

8.585.2.2 uint8_t unpack_qos_swiQosFlow_t::index

8.585.2.3 uint8_t unpack_qos_swiQosFlow_t::is_DataRate_Available

8.585.2.4 uint8_t unpack_qos_swiQosFlow_t::is_Jitter_Available

8.585.2.5 uint8_t unpack_qos_swiQosFlow_t::is_Latency_Available

8.585.2.6 uint8_t unpack_qos_swiQosFlow_t::is_LteQci_Available

8.585.2.7 uint8_t unpack_qos_swiQosFlow_t::is_MaxAllowedPktSz_Available

8.585.2.8 uint8_t unpack_qos_swiQosFlow_t::is_MinPolicedPktSz_Available

8.585.2.9 uint8_t unpack_qos_swiQosFlow_t::is_PktErrRate_Available

8.585.2.10 uint8_t unpack_qos_swiQosFlow_t::is_ProfileId3GPP2_Available

8.585.2.11 uint8_t unpack_qos_swiQosFlow_t::is_TokenBucket_Available

8.585.2.12 uint8_t unpack_qos_swiQosFlow_t::is_TrafficClass_Available

8.585.2.13 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPP2Pri_Available

8.585.2.14 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPImCn_Available

8.585.2.15 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPResResidualBER_Available

8.585.2.16 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPSigInd_Available

8.585.2.17 uint8_t unpack_qos_swiQosFlow_t::is_val_3GPPTraHdlPri_Available

8.585.2.18 uint32_t unpack_qos_swiQosFlow_t::Jitter

8.585.2.19 uint32_t unpack_qos_swiQosFlow_t::Latency

8.585.2.20 uint8_t unpack_qos_swiQosFlow_t::LteQci

8.585.2.21 uint32_t unpack_qos_swiQosFlow_t::MaxAllowedPktSz

8.585.2.22 uint32_t unpack_qos_swiQosFlow_t::MinPolicedPktSz

8.585.2.23 unpack_qos_pktErrRate_t unpack_qos_swiQosFlow_t::PktErrRate

8.585.2.24 uint16_t unpack_qos_swiQosFlow_t::ProfileId3GPP2

8.585.2.25 unpack_qos_tokenBucket_t unpack_qos_swiQosFlow_t::TokenBucket

8.585.2.26 uint8_t unpack_qos_swiQosFlow_t::TrafficClass

8.585.2.27 uint8_t unpack_qos_swiQosFlow_t::val_3GPP2Pri

8.585.2.28 uint8_t unpack_qos_swiQosFlow_t::val_3GPPImCn

8.585.2.29 uint16_t unpack_qos_swiQosFlow_t::val_3GPPResResidualBER

8.585.2.30 uint8_t unpack_qos_swiQosFlow_t::val_3GPPSigInd

8.585.2.31 uint8_t unpack_qos_swiQosFlow_t::val_3GPPTraHdlPri

8.586 unpack_qos_tokenBucket_t Struct Reference

Data Fields

- uint32_t peakRate
- uint32_t tokenRate
- uint32_t bucketSz

8.586.1 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

8.586.2 Field Documentation

8.586.2.1 uint32_t unpack_qos_tokenBucket_t::bucketSz

8.586.2.2 uint32_t unpack_qos_tokenBucket_t::peakRate

8.586.2.3 uint32_t unpack_qos_tokenBucket_t::tokenRate

8.587 unpack_qos_Tos_t Struct Reference

Data Fields

- uint8_t val
- uint8_t mask

8.587.1 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.

8.587.2 Field Documentation

8.587.2.1 uint8_t unpack_qos_Tos_t::mask

8.587.2.2 uint8_t unpack_qos_Tos_t::val

8.588 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

8.588.1 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

8.588.2 Field Documentation

8.588.2.1 uint32_t unpack_QosFlowStat_t::bearerId

8.588.2.2 uint64_t unpack_QosFlowStat_t::tx_bytes

8.588.2.3 uint64_t unpack_QosFlowStat_t::tx_bytes_drp

8.588.2.4 uint32_t unpack_QosFlowStat_t::tx_pkt

8.588.2.5 uint32_t unpack_QosFlowStat_t::tx_pkt_drp

8.589 unpack_rms_GetSMSWake_t Struct Reference

Data Fields

- uint32_t enabled
- uint32_t wake_mask

8.589.1 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)

8.589.2 Field Documentation

8.589.2.1 uint32_t unpack_rms_GetSMSWake_t::enabled

8.589.2.2 uint32_t unpack_rms_GetSMSWake_t::wake_mask

8.590 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

8.590.1 Detailed Description

WDS Pkt 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.

8.590.2 Field Documentation

8.590.2.1 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxDropConutTlv

8.590.2.2 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::RxOkByteCountTlv

8.590.2.3 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::RxOkConutTlv

8.590.2.4 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxDropConutTlv

8.590.2.5 wds_DataULongLongTlv unpack_RMTransferStatistics_ind_t::TxOkByteCountTlv

8.590.2.6 wds_DataULongTlv unpack_RMTransferStatistics_ind_t::TxOkConutTlv

8.591 unpack_sar_SLQSGetRfSarState_t Struct Reference

Data Fields

- uint32_t * pSarRFState

8.591.1 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

8.591.2 Field Documentation

8.591.2.1 uint32_t* unpack_sar_SLQSGetRfSarState_t::pSarRFState

8.592 unpack_sms_GetSMSCAddress_t Struct Reference

Data Fields

- uint8_t addressSize
- uint8_t * pSMSCAddress
- uint8_t typeSize
- uint8_t * pSMSCType

8.592.1 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.

8.592.2 Field Documentation

8.592.2.1 uint8_t unpack_sms_GetSMSCAddress_t::addressSize

8.592.2.2 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCAddress

8.592.2.3 uint8_t* unpack_sms_GetSMSCAddress_t::pSMSCType

8.592.2.4 uint8_t unpack_sms_GetSMSCAddress_t::typeSize

8.593 unpack_sms_SaveSMS_t Struct Reference

Data Fields

- uint32_t * pMessageIndex

8.593.1 Detailed Description

Parameters

pMessageIndex	• The message index assigned by the device

8.593.2 Field Documentation

8.593.2.1 uint32_t* unpack_sms_SaveSMS_t::pMessageIndex

8.594 unpack_sms_SendSMS_t Struct Reference

Data Fields

- uint16_t messageID
- uint32_t messageFailureCode

8.594.1 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

8.594.2 Field Documentation

8.594.2.1 uint32_t unpack_sms_SendSMS_t::messageFailureCode

8.594.2.2 uint16_t unpack_sms_SendSMS_t::messageID

8.595 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

8.595.1 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

8.595.2 Field Documentation

8.595.2.1 struct eTWSPLMNInfoTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSPLMNTlv

8.595.2.2 struct sMSEtwsMessageTlv unpack_sms_SetNewSMSCallback_ind_t::ETWSTlv

8.595.2.3 struct sMSOnIMSTlv unpack_sms_SetNewSMSCallback_ind_t::IMSTlv

8.595.2.4 struct messageModeTlv unpack_sms_SetNewSMSCallback_ind_t::MMTlv

8.595.2.5 struct newMTMessageTlv unpack_sms_SetNewSMSCallback_ind_t::NewMMTlv

8.595.2.6 struct sMSCAddressTlv unpack_sms_SetNewSMSCallback_ind_t::SMSCTlv

8.595.2.7 struct transferRouteMessageTlv unpack_sms_SetNewSMSCallback_ind_t::TRMessageTlv

8.596 unpack_sms_SetNewSMSCallback_t Struct Reference

8.597 unpack_sms_SLQSDeleteSMS_t Struct Reference

8.598 unpack_sms_SLQSGetIndicationRegister_t Struct Reference

Data Fields

- sms_getIndicationReg * pGetIndicationRegInfo

8.598.1 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

8.598.2 Field Documentation

8.598.2.1 sms_getIndicationReg* unpack_sms_SLQSGetIndicationRegister_t::pGetIndicationRegInfo

8.599 unpack_sms_SLQSGetMessageWaiting_t Struct Reference

Data Fields

- sms_getMsgWaitingInfo * pGetMsgWaitingInfoResp

8.599.1 Detailed Description

Structure contain Parameter that provide information about the message waiting information.

Parameters

pGetMsg-WaitingInfoResp	• Pointer to structure of getMsgWaitingInfoResp – See sms_getMsgWaitingInfoResp for more information

8.599.2 Field Documentation

8.599.2.1 sms_getMsgWaitingInfo* unpack_sms_SLQSGetMessageWaiting_t::pGetMsgWaitingInfoResp

8.600 unpack_sms_SLQSGetSMS_t Struct Reference

Data Fields

- uint32_t messageTag
- uint32_t messageFormat
- uint32_t messageSize
- uint8_t message [2048]

8.600.1 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

8.600.2 Field Documentation

8.600.2.1 uint8_t unpack_sms_SLQSGetSMS_t::message[2048]

8.600.2.2 uint32_t unpack_sms_SLQSGetSMS_t::messageFormat

8.600.2.3 uint32_t unpack_sms_SLQSGetSMS_t::messageSize

8.600.2.4 uint32_t unpack_sms_SLQSGetSMS_t::messageTag

8.601 unpack_sms_SLQSGetSmsBroadcastConfig_t Struct Reference

Data Fields

- sms_qaQmi3GPPBroadcastCfgInfo * pBroadcastConfig

- sms_qaQmi3GPP2BroadcastCfgInfo * pCDMABroadcastConfig

8.601.1 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).

8.601.2 Field Documentation

8.601.2.1 sms_qaQmi3GPPBroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pBroadcastConfig

8.601.2.2 sms_qaQmi3GPP2BroadcastCfgInfo* unpack_sms_SLQSGetSmsBroadcastConfig_t::pCDMABroadcastConfig

8.602 unpack_sms_SLQSGetSMSList_t Struct Reference

Data Fields

- uint32_t messageListSize
- qmiSmsMessageList messageList [255]

8.602.1 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

8.602.2 Field Documentation

8.602.2.1 qmiSmsMessageList unpack_sms_SLQSGetSMSList_t::messageList[255]

8.602.2.2 uint32_t unpack_sms_SLQSGetSMSList_t::messageListSize

8.603 unpack_sms_SLQSGetTransLayerInfo_t Struct Reference

Data Fields

- sms_getTransLayerInfo * pGetTransLayerInfo

8.603.1 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

8.603.2 Field Documentation

8.603.2.1 sms_getTransLayerInfo∗ unpack_sms_SLQSGetTransLayerInfo_t::pGetTransLayerInfo

8.604 unpack_sms_SLQSGetTransNWRegInfo_t Struct Reference

Data Fields

- sms_getTransNWRegInfo ∗ pGetTransNWRegInfo

8.604.1 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

8.604.2 Field Documentation

8.604.2.1 sms_getTransNWRegInfo∗ unpack_sms_SLQSGetTransNWRegInfo_t::pGetTransNWRegInfo

8.605 unpack_sms_SLQSModifySMSStatus_t Struct Reference

8.606 unpack_sms_SLQSSmsGetMaxStorageSize_t Struct Reference

Data Fields

- sms_maxStorageSizeResp ∗ pMaxStorageSizeResp

8.606.1 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

8.606.2 Field Documentation

8.606.2.1 sms_maxStorageSizeResp∗ unpack_sms_SLQSSmsGetMaxStorageSize_t::pMaxStorageSizeResp

8.607 unpack_sms_SLQSSmsGetMessageProtocol_t Struct Reference

Data Fields

- sms_msgProtocolResp ∗ pMessageProtocol

8.607.1 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

8.607.2 Field Documentation

8.607.2.1 sms_msgProtocolResp∗ unpack_sms_SLQSSmsGetMessageProtocol_t::pMessageProtocol

8.608 unpack_sms_SLQSSwiGetSMSStorage_t Struct Reference

Data Fields

- uint32_t ∗ pSmsStorage

8.608.1 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

8.608.2 Field Documentation

8.608.2.1 uint32_t* unpack_sms_SLQSSwiGetSMSStorage_t::pSmsStorage

8.609 unpack_sms_SLQSWmsMemoryFullCallBack_ind_t Struct Reference

Data Fields

- uint8_t storageType
- uint8_t messageMode

8.609.1 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)

8.609.2 Field Documentation

8.609.2.1 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::messageMode

8.609.2.2 uint8_t unpack_sms_SLQSWmsMemoryFullCallBack_ind_t::storageType

8.610 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

8.610.1 Detailed Description

Structure that contains the session type response for AvMS Set Settings command

Parameters

resultcode	- Tlv Result Code.

8.610.2 Field Documentation

8.610.2.1 uint8_t unpack_swiavms_SLQSAvMSGetSettings_t::AutoConnect

8.610.2.2 uint8_t unpack_swiavms_SLQSAvMSGetSettings_t::AutoReboot

8.610.2.3 uint8_t unpack_swiavms_SLQSAvMSGetSettings_t::FwAutodownload

8.610.2.4 uint8_t unpack_swiavms_SLQSAvMSGetSettings_t::FwAutoSDM

8.610.2.5 uint8_t unpack_swiavms_SLQSAvMSGetSettings_t::FwAutoUpdate

8.610.2.6 uint32_t unpack_swiavms_SLQSAvMSGetSettings_t::OMADMEnabled

8.610.2.7 PackSwiAVMSSettingsAPNInfo* unpack_swiavms_SLQSAvMSGetSettings_t::pAPNInfo

8.610.2.8 PackSwiAVMSSettingsConnectionRetryTimers* unpack_swiavms_SLQSAvMSGetSettings_t::pConnection-RetryTimers

8.610.2.9 uint8_t* unpack_swiavms_SLQSAvMSGetSettings_t::pNotificationStore

8.610.2.10 PackSwiAVMSSettingsPeriodsInfo* unpack_swiavms_SLQSAvMSGetSettings_t::pPeroidsInfo

8.610.2.11 uint32_t* unpack_swiavms_SLQSAvMSGetSettings_t::pPollingTimer

8.610.2.12 uint32_t unpack_swiavms_SLQSAvMSGetSettings_t::resultcode

8.611 unpack_swiavms_SLQSAvMSSendSelection_t Struct Reference

Data Fields

- uint32_t resultcode

8.611.1 Detailed Description

Structure that contains the session type response for AvMS Selection command

Parameters

resultcode	- Tlv Result Code.

8.611.2 Field Documentation

8.611.2.1 uint32_t unpack_swiavms_SLQSAvMSSendSelection_t::resultcode

8.612 unpack_swiavms_SLQSAvmsSetEventReport_t Struct Reference

Data Fields

- uint32_t resultcode

8.612.1 Detailed Description

Structure that contains the session type response for AvMS set event report command

Parameters

resultcode	- Tlv Result Code.

8.612.2 Field Documentation

8.612.2.1 uint32_t unpack_swiavms_SLQSAvmsSetEventReport_t::resultcode

8.613 unpack_swiavms_SLQSAvMSSetSettings_t Struct Reference

Data Fields

- uint32_t resultcode

8.613.1 Detailed Description

Structure that contains the session type response for AvMS Set Settings command

Parameters

resultcode	- Tlv Result Code.

8.613.2 Field Documentation

8.613.2.1 uint32_t unpack_swiavms_SLQSAvMSSetSettings_t::resultcode

8.614 unpack_swiavms_SLQSAvMSStartSession_t Struct Reference

Data Fields

- uint32_t sessionResponse
- uint32_t resultcode

8.614.1 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.

8.614.2 Field Documentation

8.614.2.1 uint32_t unpack_swiavms_SLQSAvMSStartSession_t::resultcode

8.614.2.2 uint32_t unpack_swiavms_SLQSAvMSStartSession_t::sessionResponse

8.615 unpack_swiavms_SLQSAvMSStopSession_t Struct Reference

Data Fields

- uint32_t resultcode

8.615.1 Detailed Description

Structure that contains the session type response for AvMS Stop session command

Parameters

resultcode	- Tlv Result Code.

8.615.2 Field Documentation

8.615.2.1 uint32_t unpack_swiavms_SLQSAvMSStopSession_t::resultcode

8.616 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

8.616.1 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

8.616.2 Field Documentation

8.616.2.1 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::fix_rate

8.616.2.2 int unpack_swiloc_SwiLocGetAutoStart_t::fix_rate_reported

8.616.2.3 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::fix_type

8.616.2.4 int unpack_swiloc_SwiLocGetAutoStart_t::fix_type_reported

8.616.2.5 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::function

8.616.2.6 int unpack_swiloc_SwiLocGetAutoStart_t::function_reported

8.616.2.7 uint32_t unpack_swiloc_SwiLocGetAutoStart_t::max_dist

8.616.2.8 int unpack_swiloc_SwiLocGetAutoStart_t::max_dist_reported

8.616.2.9 uint8_t unpack_swiloc_SwiLocGetAutoStart_t::max_time

8.616.2.10 int unpack_swiloc_SwiLocGetAutoStart_t::max_time_reported

8.617 unpack_swioma_SLQSOMADMAlertCallback_ind_t Struct Reference

Data Fields

- uint32_t eventType
- unpack_omaDmFotaTlv_t SessionInfoFota
- unpack_omaDmConfigTlv_t SessionInfoConfig
- unpack_omaDmNotificationsTlv_t SessionInfoNotification

8.617.1 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

8.617.2 Field Documentation

8.617.2.1 uint32_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::eventType

8.617.2.2 unpack_omaDmConfigTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoConfig

8.617.2.3 unpack_omaDmFotaTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoFota

8.617.2.4 unpack_omaDmNotificationsTlv_t unpack_swioma_SLQSOMADMAlertCallback_ind_t::SessionInfoNotification

8.618 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

8.618.1 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

8.618.2 Field Documentation

8.618.2.1 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Date[255]

8.618.2.2 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::DateLength

8.618.2.3 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescLength

8.618.2.4 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgDescription[255]

8.618.2.5 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgName[255]

8.618.2.6 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::PkgNameLength

8.618.2.7 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::RetryCount

8.618.2.8 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionState

8.618.2.9 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SessionType

8.618.2.10 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Severity

8.618.2.11 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Source[255]

8.618.2.12 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::SourceLength

8.618.2.13 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Status

8.618.2.14 uint8_t unpack_swioma_SLQSOMADMGetSessionInfo_t::Time[255]

8.618.2.15 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::TimeLength

8.618.2.16 uint16_t unpack_swioma_SLQSOMADMGetSessionInfo_t::UpdateCompleteStatus

8.619 unpack_swioma_SLQSOMADMGetSettings_t Struct Reference

Data Fields

- uint32_t OMADMEnabled
- uint8_t FOTAdownload
- uint8_t FOTAUpdate
- uint8_t Autosdm
- uint8_t FwAutoCheck

8.619.1 Detailed Description

Structure containing the OMA DM settings retrieved from the device

Parameters

OMADM-Enabled[OUT]	• 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.

FOTAdownload[-OUT]	• 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.
FOTAUpdate[OUT]	• 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.
Autosdm[OUT]	• 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.
FwAutoCheck[OUT]	• 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.

8.619.2 Field Documentation

8.619.2.1 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::Autosdm

8.619.2.2 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAdownload

8.619.2.3 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FOTAUpdate

8.619.2.4 uint8_t unpack_swioma_SLQSOMADMGetSettings_t::FwAutoCheck

8.619.2.5 uint32_t unpack_swioma_SLQSOMADMGetSettings_t::OMADMEnabled

8.620 unpack_swioma_SLQSOMADMStartSession_t Struct Reference

Data Fields

- uint32_t FwAvailability

8.620.1 Detailed Description

Structure that contains the responses for OMA start session command

Parameters

pFwAvailability[-OUT]	• 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

8.620.2 Field Documentation

8.620.2.1 uint32_t unpack_swioma_SLQSOMADMStartSession_t::FwAvailability

8.621 unpack_uim_ChangePin_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t TlvResult

8.621.1 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.

8.621.2 Field Documentation

8.621.2.1 uim_encryptedPIN1* unpack_uim_ChangePin_t::pEncryptedPIN1

8.621.2.2 uint32_t* unpack_uim_ChangePin_t::pIndicationToken

8.621.2.3 uim_remainingRetries* unpack_uim_ChangePin_t::pRemainingRetries

8.621.2.4 uint16_t unpack_uim_ChangePin_t::Tlvresult

8.622 unpack_uim_GetCardStatus_t Struct Reference

Data Fields

- uim_cardStatus * pCardStatus
- uim_hotSwapStatus * pHotSwapStatus
- uint16_t Tlvresult

8.622.1 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.

8.622.2 Field Documentation

8.622.2.1 uim_cardStatus* unpack_uim_GetCardStatus_t::pCardStatus

8.622.2.2 uim_hotSwapStatus* unpack_uim_GetCardStatus_t::pHotSwapStatus

8.622.2.3 uint16_t unpack_uim_GetCardStatus_t::Tlvresult

8.623 unpack_uim_ReadTransparent_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_readResult * pReadResult
- uint32_t * pIndicationToken
- uint8_t * pEncryptedData
- uint16_t Tlvresult

8.623.1 Detailed Description

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

Parameters

pCardResult	• See cardResult for more information.
pReadResult	• See 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.

8.623.2 Field Documentation

8.623.2.1 uim_cardResult* unpack_uim_ReadTransparent_t::pCardResult

8.623.2.2 uint8_t* unpack_uim_ReadTransparent_t::pEncryptedData

8.623.2.3 uint32_t* unpack_uim_ReadTransparent_t::pIndicationToken

8.623.2.4 uim_readResult* unpack_uim_ReadTransparent_t::pReadResult

8.623.2.5 uint16_t unpack_uim_ReadTransparent_t::Tlvresult

8.624 unpack_uim_SetPinProtection_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries
- uim_encryptedPIN1 * pEncryptedPIN1
- uint32_t * pIndicationToken
- uint16_t Tlvresult

8.624.1 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.

8.624.2 Field Documentation

8.624.2.1 uim_encryptedPIN1* unpack_uim_SetPinProtection_t::pEncryptedPIN1

8.624.2.2 uint32_t* unpack_uim_SetPinProtection_t::pIndicationToken

8.624.2.3 uim_remainingRetries* unpack_uim_SetPinProtection_t::pRemainingRetries

8.624.2.4 uint16_t unpack_uim_SetPinProtection_t::TlvResult

8.625 unpack_uim_SetUimSlotStatusChangeCallback_ind_t Struct Reference

Data Fields

- slots_t slotsstatusChange
- uint8_t bNumberOfPhySlots

8.625.1 Detailed Description

Structure consist of card status params

Parameters

slotsstatus-Change	• See slot_t for more information
bNumberOfPhy-Slots	• Number of Physical Slot(s)

8.625.2 Field Documentation

8.625.2.1 uint8_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::bNumberOfPhySlots

8.625.2.2 slots_t unpack_uim_SetUimSlotStatusChangeCallback_ind_t::slotsstatusChange

8.626 unpack_uim_SLQSUIMAuthenticate_t Struct Reference

Data Fields

- uim_cardResult * pCardResult

- uim_authenticateResult * pAuthenticateResult
- uint32_t * pIndicationToken

8.626.1 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.

8.626.2 Field Documentation

8.626.2.1 uim_authenticateResult* unpack_uim_SLQSUIMAuthenticate_t::pAuthenticateResult

8.626.2.2 uim_cardResult* unpack_uim_SLQSUIMAuthenticate_t::pCardResult

8.626.2.3 uint32_t* unpack_uim_SLQSUIMAuthenticate_t::pIndicationToken

8.627 unpack_uim_SLQSUIMDepersonalization_t Struct Reference

Data Fields

- uim_remainingRetries * pRemainingRetries

8.627.1 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.

8.627.2 Field Documentation

8.627.2.1 uim_remainingRetries* unpack_uim_SLQSUIMDepersonalization_t::pRemainingRetries

8.628 unpack_uim_SLQSUIMEventRegister_t Struct Reference

Data Fields

- uint32_t eventMask

8.628.1 Detailed Description

Parameters

eventMask	- bit 0 - card status • bit 1 - SAP connection • bit 4 - physical slot status

8.628.2 Field Documentation

8.628.2.1 uint32_t unpack_uim_SLQSUIMEventRegister_t::eventMask

8.629 unpack_uim_SLQSUIMGetConfiguration_t Struct Reference

Data Fields

- uint8_t * pAutoSelection
- uim_personalizationStatus * pPersonalizationStatus
- uint8_t * pHaltSubscription

8.629.1 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

8.629.2 Field Documentation

8.629.2.1 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pAutoSelection

8.629.2.2 uint8_t* unpack_uim_SLQSUIMGetConfiguration_t::pHaltSubscription

8.629.2.3 uim_personalizationStatus* unpack_uim_SLQSUIMGetConfiguration_t::pPersonalizationStatus

8.630 unpack_uim_SLQSUIMGetFileAttributes_t Struct Reference

Data Fields

- uim_cardResult * pCardResult
- uim_fileAttributes * pFileAttributes
- uint32_t * pIndicationToken

8.630.1 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.

8.630.2 Field Documentation

8.630.2.1 uim_cardResult* unpack_uim_SLQSUIMGetFileAttributes_t::pCardResult

8.630.2.2 uim_fileAttributes* unpack_uim_SLQSUIMGetFileAttributes_t::pFileAttributes

8.630.2.3 uint32_t* unpack_uim_SLQSUIMGetFileAttributes_t::pIndicationToken

8.631 unpack_uim_SLQSUIMGetSlotsStatus_t Struct Reference

Data Fields

- uint8_t * pNumberOfPhySlot
- slots_t * pUimSlotsStatus

8.631.1 Detailed Description

This structure contains information of the response parameters associated with a Get Slots Status API.

Parameters

pNumberOfPhy-Slot	• Number of sets of the Slot Status.
pUimSlotsStatus	• Slots Status See slots_t for more information..

8.631.2 Field Documentation

8.631.2.1 uint8_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pNumberOfPhySlot

8.631.2.2 slots_t* unpack_uim_SLQSUIMGetSlotsStatus_t::pUimSlotsStatus

8.632 unpack_uim_SLQSUIMRefreshGetLastEvent_t Struct Reference

Data Fields

- uim_refreshevent * pRefreshEvent

8.632.1 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.

8.632.2 Field Documentation

8.632.2.1 uim_refreshevent* unpack_uim_SLQSUIMRefreshGetLastEvent_t::pRefreshEvent

8.633 unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t Struct Reference

Data Fields

- uim_cardStatus * pCardStatus

8.633.1 Detailed Description

This structure contains information about Status change callback.

Parameters

pCardStatus	Card Status • See uim_cardStatus for more information.

8.633.2 Field Documentation

8.633.2.1 uim_cardStatus∗ unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t::pCardStatus

8.634 unpack_uim_UnblockPin_t Struct Reference

Data Fields

- uim_remainingRetries ∗ pRemainingRetries
- uim_encryptedPIN1 ∗ pEncryptedPIN1
- uint32_t ∗ pIndicationToken
- uint16_t TlvresultTlvresult

8.634.1 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

8.634.2 Field Documentation

8.634.2.1 uim_encryptedPIN1∗ unpack_uim_UnblockPin_t::pEncryptedPIN1

8.634.2.2 uint32_t∗ unpack_uim_UnblockPin_t::pIndicationToken

8.634.2.3 uim_remainingRetries∗ unpack_uim_UnblockPin_t::pRemainingRetries

8.634.2.4 uint16_t unpack_uim_UnblockPin_t::Tlvresult

8.635 unpack_uim_VerifyPin_t Struct Reference

Data Fields

- uim_remainingRetries ∗ pRemainingRetries
- uim_encryptedPIN1 ∗ pEncryptedPIN1
- uint32_t ∗ pIndicationToken
- uint16_t TlvresultTlvresult

8.635.1 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.

8.635.2 Field Documentation

8.635.2.1 uim_encryptedPIN1∗ unpack_uim_VerifyPin_t::pEncryptedPIN1

8.635.2.2 uint32_t∗ unpack_uim_VerifyPin_t::pIndicationToken

8.635.2.3 uim_remainingRetries∗ unpack_uim_VerifyPin_t::pRemainingRetries

8.635.2.4 uint16_t unpack_uim_VerifyPin_t::TlvResult

8.636 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

8.636.1 Field Documentation

8.636.1.1 voice_alphaIDInfo∗ unpack_voice_SLQSOriginateUSSD_t::pAlphaIDInfo

8.636.1.2 uint8_t∗ unpack_voice_SLQSOriginateUSSD_t::pCallId

8.636.1.3 uint8_t∗ unpack_voice_SLQSOriginateUSSD_t::pCcResultType

8.636.1.4 voice_ccSUPSType∗ unpack_voice_SLQSOriginateUSSD_t::pCCSuppsType

8.636.1.5 uint16_t∗ unpack_voice_SLQSOriginateUSSD_t::pfailureCause

8.636.1.6 struct voice_USSInfo∗ unpack_voice_SLQSOriginateUSSD_t::pUSSDInfo

8.637 unpack_voice_SLQSVoiceAnswerCall_t Struct Reference

Data Fields

- uint8_t * pCallId

8.637.1 Detailed Description

Contains the parameters passed for pack voice Answer Call.

Parameters

pCallId	• Unique call identifier for the call that must be answered.

8.637.2 Field Documentation

8.637.2.1 uint8_t* unpack_voice_SLQSVoiceAnswerCall_t::pCallId

8.638 unpack_voice_SLQSVoiceBurstDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID

8.638.1 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

8.638.2 Field Documentation

8.638.2.1 uint8_t* unpack_voice_SLQSVoiceBurstDTMF_t::pCallID

8.639 unpack_voice_SLQSVoiceDialCall_t Struct Reference

Data Fields

- uint8_t * pCallID
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResultType
- voice_ccSUPSType * pCCSUPSType

8.639.1 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

8.639.2 Field Documentation

8.639.2.1 voice_alphaIDInfo∗ unpack_voice_SLQSVoiceDialCall_t::pAlphaIDInfo

8.639.2.2 uint8_t∗ unpack_voice_SLQSVoiceDialCall_t::pCallID

8.639.2.3 uint8_t∗ unpack_voice_SLQSVoiceDialCall_t::pCCResultType

8.639.2.4 voice_ccSUPSType∗ unpack_voice_SLQSVoiceDialCall_t::pCCSUPSType

8.640 unpack_voice_SLQSVoiceEndCall_t Struct Reference

Data Fields

- uint8_t ∗ pCallId

8.640.1 Detailed Description

Parameters

pCallId	• Unique call identifier for the call that must be ended

8.640.2 Field Documentation

8.640.2.1 uint8_t∗ unpack_voice_SLQSVoiceEndCall_t::pCallId

8.641 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

8.641.1 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.

pOTASP-Status(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
pVoice-Privacy(optional)	• Voice Privacy. • Values. – 0x00 - VOICE_PRIVACY_STANDARD - Standard privacy – 0x01 - VOICE_PRIVACY_ENHANCED - Enhanced privacy – 0xFF - Not Available
pArrCallEnd-Reason(optional)	• See voice_arrCallEndReason for more information.
pArrAlphaI-D(optional)	• See voice_arrAlphaID for more information.
pArrConnect-Party-Num(optional)	• See voice_arrConnectPartyNum for more information.
pArrDiag-Info(optional)	• See voice_arrDiagInfo for more information.
pArrCalledParty-Num(optional)	• See voice_arrCalledPartyNum for more information.

pArrRedirParty-Num(optional)	• See voice_arrRedirPartyNum for more information.
pArrAlerting-Pattern(optional)	• See voice_arrAlertingPattern for more information.

8.641.2 Field Documentation

8.641.2.1 voice_arrAlertingPattern∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingPattern

8.641.2.2 voice_arrAlertingType∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlertingType

8.641.2.3 voice_arrAlphaID∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrAlphaID

8.641.2.4 voice_arrCalledPartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCalledPartyNum

8.641.2.5 voice_arrCallEndReason∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallEndReason

8.641.2.6 voice_arrCallInfo∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrCallInfo

8.641.2.7 voice_arrConnectPartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrConnectPartyNum

8.641.2.8 voice_arrDiagInfo∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrDiagInfo

8.641.2.9 voice_arrRedirPartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRedirPartyNum

8.641.2.10 voice_arrRemotePartyName∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyName

8.641.2.11 voice_arrRemotePartyNum∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrRemotePartyNum

8.641.2.12 voice_arrSvcOption∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrSvcOption

8.641.2.13 voice_arrUUSInfo∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pArrUUSInfo

8.641.2.14 uint8_t∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pOTASPStatus

8.641.2.15 uint8_t∗ unpack_voice_SLQSVoiceGetAllCallInfo_t::pVoicePrivacy

8.642 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

8.642.1 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.

8.642.2 Field Documentation

8.642.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallBarring_t::pAlphaIDInfo

8.642.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCallID

8.642.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pCCResType

8.642.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallBarring_t::pCCSUPSType

8.642.2.5 uint16_t* unpack_voice_SLQSVoiceGetCallBarring_t::pFailCause

8.642.2.6 uint8_t* unpack_voice_SLQSVoiceGetCallBarring_t::pSvcClass

8.643 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

8.643.1 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.

8.643.2 Field Documentation

8.643.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pAlphaIDInfo

8.643.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCallID

8.643.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCResType

8.643.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pCCSUPSType

8.643.2.5 uint16_t* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pFailCause

8.643.2.6 voice_getCallFWExtInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWExtInfo

8.643.2.7 voice_getCallFWInfo* unpack_voice_SLQSVoiceGetCallForwardingStatus_t::pGetCallFWInfo

8.644 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

8.644.1 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 vocie_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

8.644.2 Field Documentation

8.644.2.1 uint32_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertingPattern

8.644.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pAlertType

8.644.2.3 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pAlphaIDInfo

8.644.2.4 voice_callInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pCallInfo

8.644.2.5 voice_connectNumInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pConnectNumInfo

8.644.2.6 voice_diagInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pDiagInfo

8.644.2.7 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pOTASPStatus

8.644.2.8 voice_remotePartyName* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyName

8.644.2.9 voice_remotePartyNum* unpack_voice_SLQSVoiceGetCallInfo_t::pRemotePartyNum

8.644.2.10 uint16_t* unpack_voice_SLQSVoiceGetCallInfo_t::pSrvOpt

8.644.2.11 voice_UUSInfo* unpack_voice_SLQSVoiceGetCallInfo_t::pUUSInfo

8.644.2.12 uint8_t* unpack_voice_SLQSVoiceGetCallInfo_t::pVoicePrivacy

8.645 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

8.645.1 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.

8.645.2 Field Documentation

8.645.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCallWaiting_t::pAlphaIDInfo

8.645.2.2 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCallID

8.645.2.3 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCResType

8.645.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCallWaiting_t::pCCSUPSType

8.645.2.5 uint16_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pFailCause

8.645.2.6 uint8_t* unpack_voice_SLQSVoiceGetCallWaiting_t::pSvcClass

8.646 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

8.646.1 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.

8.646.2 Field Documentation

8.646.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIP_t::pAlphaIDInfo

8.646.2.2 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCallID

8.646.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIP_t::pCCResType

8.646.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIP_t::pCCSUPSType

8.646.2.5 voice_CLIPResp* unpack_voice_SLQSVoiceGetCLIP_t::pCLIPResp

8.646.2.6 uint16_t* unpack_voice_SLQSVoiceGetCLIP_t::pFailCause

8.647 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

8.647.1 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.

8.647.2 Field Documentation

8.647.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCLIR_t::pAlphaIDInfo

8.647.2.2 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCallID

8.647.2.3 uint8_t* unpack_voice_SLQSVoiceGetCLIR_t::pCCResType

8.647.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCLIR_t::pCCSUPSType

8.647.2.5 voice_CLIRResp* unpack_voice_SLQSVoiceGetCLIR_t::pCLIRResp

8.647.2.6 uint16_t* unpack_voice_SLQSVoiceGetCLIR_t::pFailCause

8.648 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

8.648.1 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.

8.648.2 Field Documentation

8.648.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCNAP_t::pAlphaIDInfo

8.648.2.2 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCallID

8.648.2.3 uint8_t* unpack_voice_SLQSVoiceGetCNAP_t::pCCResType

8.648.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCNAP_t::pCCSUPSType

8.648.2.5 voice_CNAPResp* unpack_voice_SLQSVoiceGetCNAP_t::pCNAPResp

8.648.2.6 uint16_t* unpack_voice_SLQSVoiceGetCNAP_t::pFailCause

8.649 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

8.649.1 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.

8.649.2 Field Documentation

8.649.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLP_t::pAlphaIDInfo

8.649.2.2 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCallID

8.649.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLP_t::pCCResType

8.649.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLP_t::pCCSUPSType

8.649.2.5 voice_COLPResp* unpack_voice_SLQSVoiceGetCOLP_t::pCOLPResp

8.649.2.6 uint16_t* unpack_voice_SLQSVoiceGetCOLP_t::pFailCause

8.650 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

8.650.1 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.

8.650.2 Field Documentation

8.650.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceGetCOLR_t::pAlphaIDInfo

8.650.2.2 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCallID

8.650.2.3 uint8_t* unpack_voice_SLQSVoiceGetCOLR_t::pCCResType

8.650.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceGetCOLR_t::pCCSUPSType

8.650.2.5 voice_COLRResp* unpack_voice_SLQSVoiceGetCOLR_t::pCOLRResp

8.650.2.6 uint16_t* unpack_voice_SLQSVoiceGetCOLR_t::pFailCause

8.651 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

8.651.1 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 airTimer for more information
pRoamTimer-Cnt(optional)	• Roam Timer Count • Value returned is read from NV_ROAM_CNT_I. • See 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.

8.651.2 Field Documentation

8.651.2.1 voice_airTimer* unpack_voice_SLQSVoiceGetConfig_t::pAirTimerCnt

8.651.2.2 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pAutoAnswerStat

8.651.2.3 voice_curAMRConfig* unpack_voice_SLQSVoiceGetConfig_t::pCurAMRConfig

8.651.2.4 voice_prefVoiceSO* unpack_voice_SLQSVoiceGetConfig_t::pCurPrefVoiceSO

8.651.2.5 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurrTTYMode

8.651.2.6 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoiceDomainPref

8.651.2.7 uint8_t* unpack_voice_SLQSVoiceGetConfig_t::pCurVoicePrivacyPref

8.651.2.8 voice_roamTimer* unpack_voice_SLQSVoiceGetConfig_t::pRoamTimerCnt

8.652 unpack_voice_SLQSVoiceManageCalls_t Struct Reference

Data Fields

- uint16_t * pFailCause

8.652.1 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

8.652.2 Field Documentation

8.652.2.1 uint16_t* unpack_voice_SLQSVoiceManageCalls_t::pFailCause

8.653 unpack_voice_SLQSVoiceSendFlash_t Struct Reference

Data Fields

- uint8_t * pCallID

8.653.1 Detailed Description

This structure contains the flash information associated with a call.

Parameters

pCallID	• Unique call identifier associated with the current call.

8.653.2 Field Documentation

8.653.2.1 uint8_t* unpack_voice_SLQSVoiceSendFlash_t::pCallID

8.654 unpack_voice_SLQSVoiceSetCallBarringPassword_t Struct Reference

Data Fields

- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.654.1 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.

8.654.2 Field Documentation

8.654.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pAlphaIDInfo

8.654.2.2 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCallID

8.654.2.3 uint8_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCResType

8.654.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pCCSUPSType

8.654.2.5 uint16_t* unpack_voice_SLQSVoiceSetCallBarringPassword_t::pFailCause

8.655 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

8.655.1 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.

8.655.2 Field Documentation

8.655.2.1 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAirTimerStatus

8.655.2.2 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pAutoAnsStatus

8.655.2.3 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pPrefVoiceSOStatus

8.655.2.4 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pRoamTimerStatus

8.655.2.5 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pTTYConfigStatus

8.655.2.6 uint8_t* unpack_voice_SLQSVoiceSetConfig_t::pVoiceDomainPrefStatus

8.656 unpack_voice_SLQSVoiceSetSUPSService_t Struct Reference

Data Fields

- uint16_t * pFailCause
- voice_alphaIDInfo * pAlphaIDInfo
- uint8_t * pCCResultType
- uint8_t * pCallID
- voice_ccSUPSType * pCCSUPSType

8.656.1 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

8.656.2 Field Documentation

8.656.2.1 voice_alphaIDInfo* unpack_voice_SLQSVoiceSetSUPSService_t::pAlphaIDInfo

8.656.2.2 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCallID

8.656.2.3 uint8_t* unpack_voice_SLQSVoiceSetSUPSService_t::pCCResultType

8.656.2.4 voice_ccSUPSType* unpack_voice_SLQSVoiceSetSUPSService_t::pCCSUPSType

8.656.2.5 uint16_t* unpack_voice_SLQSVoiceSetSUPSService_t::pFailCause

8.657 unpack_voice_SLQSVoiceStartContDTMF_t Struct Reference

Data Fields

- uint8_t * pCallID

8.657.1 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

8.657.2 Field Documentation

8.657.2.1 uint8_t* unpack_voice_SLQSVoiceStartContDTMF_t::pCallID

8.658 unpack_voice_SLQSVoiceStopContDTMF_t Struct Reference

Data Fields

- uint8_t callID

8.658.1 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

8.658.2 Field Documentation

8.658.2.1 uint8_t unpack_voice_SLQSVoiceStopContDTMF_t::callID

8.659 unpack_wds_DHCPv4ClientLease_ind_t Struct Reference

Data Fields

- wds_DHCPProfileIdTlv ProfileIdTlv
- wds_DHCPLeaseStateTlv DHCPv4LeaseStateTlv
- wds_IPv4AdTlv IPv4AddrTlv
- wds_DHCPLeaseOptTlv DHCPv4LeaseOptTlv

8.659.1 Field Documentation

8.659.1.1 wds_DHCPLeaseOptTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseOptTlv

8.659.1.2 wds_DHCPLeaseStateTlv unpack_wds_DHCPv4ClientLease_ind_t::DHCPv4LeaseStateTlv

8.659.1.3 wds_IPv4AdTlv unpack_wds_DHCPv4ClientLease_ind_t::IPv4AddrTlv

8.659.1.4 wds_DHCPProfileIdTlv unpack_wds_DHCPv4ClientLease_ind_t::ProfileIdTlv

8.660 unpack_wds_GetAutoconnect_t Struct Reference

Data Fields

- uint32_t * psetting

8.660.1 Detailed Description

auto connect data session setting parameter.

Parameters

pSetting	• NDIS auto connect setting – 0 - Disabled – 1 - Enabled

8.660.2 Field Documentation

8.660.2.1 uint32_t* unpack_wds_GetAutoconnect_t::psetting

8.661 unpack_wds_GetByteTotals_t Struct Reference

Data Fields

- uint64_t * pTXTotalBytes
- uint64_t * pRXTotalBytes

8.661.1 Detailed Description

Parameters

pTXTotalBytes	• Bytes transmitted without error
pRXTotalBytes	• Bytes received without error

8.661.2 Field Documentation

8.661.2.1 uint64_t* unpack_wds_GetByteTotals_t::pRXTotalBytes

8.661.2.2 uint64_t* unpack_wds_GetByteTotals_t::pTXTotalBytes

8.662 unpack_wds_GetConnectionRate_t Struct Reference

Data Fields

- uint32_t currentChannelTXRate
- uint32_t currentChannelRXRate
- uint32_t maxChannelTXRate
- uint32_t maxChannelRXRate

8.662.1 Detailed Description

Parameters

currentChannelTXRate	Instantaneous channel Tx rate
currentChannelRXRate	Instantaneous channel Rx rate
maxChannelTXRate	Maximum Tx rate
maxChannelRXRate	Maximum Rx rate

8.662.2 Field Documentation

8.662.2.1 uint32_t unpack_wds_GetConnectionRate_t::currentChannelRXRate

8.662.2.2 uint32_t unpack_wds_GetConnectionRate_t::currentChannelTXRate

8.662.2.3 uint32_t unpack_wds_GetConnectionRate_t::maxChannelRXRate

8.662.2.4 uint32_t unpack_wds_GetConnectionRate_t::maxChannelTXRate

8.663 unpack_wds_GetDataBearerTechnology_t Struct Reference

Data Fields

- uint32_t * pDataBearer

8.663.1 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

8.663.2 Field Documentation

8.663.2.1 uint32_t* unpack_wds_GetDataBearerTechnology_t::pDataBearer

8.664 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]

8.664.1 Detailed Description

Parameters

pdptype	pdp type
ipaddr	ip address
pridns	primary dns
secdns	secondry dns
ipaddrv6	ip address v6
pridnsv6	primary dns v6
secdnsv6	secondry dns v6
namesize	profile name size
name	profile name
apnsize	apn size
apnname	apn name
usersize	username size
username	username

8.664.2 Field Documentation

8.664.2.1 int8_t unpack_wds_GetDefaultProfile_t::apnname[255]

8.664.2.2 uint8_t unpack_wds_GetDefaultProfile_t::apnsize

8.664.2.3 uint32_t unpack_wds_GetDefaultProfile_t::auth

8.664.2.4 uint32_t unpack_wds_GetDefaultProfile_t::ipaddr

8.664.2.5 uint16_t unpack_wds_GetDefaultProfile_t::ipaddrv6

8.664.2.6 int8_t unpack_wds_GetDefaultProfile_t::name[255]

8.664.2.7 uint8_t unpack_wds_GetDefaultProfile_t::namesize

8.664.2.8 uint32_t unpack_wds_GetDefaultProfile_t::pdptype

8.664.2.9 uint32_t unpack_wds_GetDefaultProfile_t::pridns

8.664.2.10 uint16_t unpack_wds_GetDefaultProfile_t::pridnsv6

8.664.2.11 uint32_t unpack_wds_GetDefaultProfile_t::secdns

8.664.2.12 uint16_t unpack_wds_GetDefaultProfile_t::secdnsv6

8.664.2.13 int8_t unpack_wds_GetDefaultProfile_t::username[255]

8.664.2.14 uint8_t unpack_wds_GetDefaultProfile_t::usersize

8.665 unpack_wds_GetDefaultProfileNum_t Struct Reference

Data Fields

- uint8_t index

8.665.1 Detailed Description

Parameters

index	profile index

8.665.2 Field Documentation

8.665.2.1 uint8_t unpack_wds_GetDefaultProfileNum_t::index

8.666 unpack_wds_GetDormancyState_t Struct Reference

Data Fields

- uint32_t dormancyState

8.666.1 Detailed Description

Parameters

dormancyState	dormancy status

8.666.2 Field Documentation

8.666.2.1 uint32_t unpack_wds_GetDormancyState_t::dormancyState

8.667 unpack_wds_GetLastMobileIPError_t Struct Reference

Data Fields

- uint32_t error

8.667.1 Detailed Description

Parameters

error	last mip status 0-success >0- error code

8.667.2 Field Documentation

8.667.2.1 uint32_t unpack_wds_GetLastMobileIPError_t::error

8.668 unpack_wds_GetMobileIP_t Struct Reference

Data Fields

- uint32_t mipMode

8.668.1 Detailed Description

Parameters

mipMode	mobile IP mode

8.668.2 Field Documentation

8.668.2.1 uint32_t unpack_wds_GetMobileIP_t::mipMode

8.669 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

8.669.1 Detailed Description

Parameters

enabled	mobile ip profile state enabled/disabled
address	mobile ip profile home address
primaryHA	mobile ip profile home agent primary
secondaryHA	mobile ip profile secondary home agent address
revTunneling	mobile ip profile rev tunneling
naiSize	mobile ip profile NAI size
NAI	NAI string in ASCII text.
HASPI	HA security parameter index.
AAASPI	AAA server security parameter index.
HAState	Mobile IP Profile HA Key State
AAAState	Mobile IP Profile AAA Key State

8.669.2 Field Documentation

8.669.2.1 uint32_t unpack_wds_GetMobileIPProfile_t::AAASPI

8.669.2.2 uint32_t unpack_wds_GetMobileIPProfile_t::AAAState

8.669.2.3 uint32_t unpack_wds_GetMobileIPProfile_t::address

8.669.2.4 uint8_t unpack_wds_GetMobileIPProfile_t::enabled

8.669.2.5 uint32_t unpack_wds_GetMobileIPProfile_t::HASPI

8.669.2.6 uint32_t unpack_wds_GetMobileIPProfile_t::HAState

8.669.2.7 int8_t unpack_wds_GetMobileIPProfile_t::NAI[255]

8.669.2.8 uint8_t unpack_wds_GetMobileIPProfile_t::naiSize

8.669.2.9 uint32_t unpack_wds_GetMobileIPProfile_t::primaryHA

8.669.2.10 uint8_t unpack_wds_GetMobileIPProfile_t::revTunneling

8.669.2.11 uint32_t unpack_wds_GetMobileIPProfile_t::secondaryHA

8.670 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

8.670.1 Detailed Description

Parameters

pTXPacketSuccesses	• No of transmitted Packets without error.
pRXPacketSuccesses	• No of received Packets without error.
pTXPacketErrors	• Number of outgoing packets with framing errors.
pRXPacketErrors	• Number of incoming packets with framing errors.
pTXPacketOverflows	• Number of packets dropped because Tx buffer overflowed (out of memory).
pRXPacketOverflows	• Number of packets dropped because Rx buffer overflowed (out of memory).
pTXOkBytesCount	• No of bytes transmitted without error.
pRXOkBytesCount	• No of bytes received without error.
pTXOKBytesLastCall	• 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

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.

8.670.2 Field Documentation

8.670.2.1 uint32_t* unpack_wds_GetPacketStatistics_t::pRXDroppedCount

8.670.2.2 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOkBytesCount

8.670.2.3 uint64_t* unpack_wds_GetPacketStatistics_t::pRXOKBytesLastCall

8.670.2.4 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketErrors

8.670.2.5 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketOverflows

8.670.2.6 uint32_t* unpack_wds_GetPacketStatistics_t::pRXPacketSuccesses

8.670.2.7 uint32_t* unpack_wds_GetPacketStatistics_t::pTXDroppedCount

8.670.2.8 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOkBytesCount

8.670.2.9 uint64_t* unpack_wds_GetPacketStatistics_t::pTXOKBytesLastCall

8.670.2.10 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketErrors

8.670.2.11 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketOverflows

8.670.2.12 uint32_t* unpack_wds_GetPacketStatistics_t::pTXPacketSuccesses

8.671 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

8.671.1 Detailed Description

Parameters

tXPacket-Successes	Tx Packets OK
rXPacket-Successes	Rx Packets OK
tXPacketErrors	Tx Packet Errors
rXPacketErrors	Rx Packet Errors
tXPacket-Overflows	Tx Overflows
rXPacket-Overflows	Rx Overflows
tXOkBytesCount	Tx Bytes OK
rXOkBytesCount	Rx Bytes OK
tXOKBytesLast-Call	Last call Tx Bytes OK
rXOKBytesLast-Call	Last call Rx Bytes OK
tXDroppedCount	Tx Packets Dropped
rXDroppedCount	Rx Packets Dropped

8.671.2 Field Documentation

8.671.2.1 uint32_t unpack_wds_GetPacketStatus_t::rXDroppedCount

8.671.2.2 uint64_t unpack_wds_GetPacketStatus_t::rXOkBytesCount

8.671.2.3 uint64_t unpack_wds_GetPacketStatus_t::rXOKBytesLastCall

8.671.2.4 uint32_t unpack_wds_GetPacketStatus_t::rXPacketErrors

8.671.2.5 uint32_t unpack_wds_GetPacketStatus_t::rXPacketOverflows

8.671.2.6 uint32_t unpack_wds_GetPacketStatus_t::rXPacketSuccesses

8.671.2.7 uint32_t unpack_wds_GetPacketStatus_t::tXDroppedCount

8.671.2.8 uint64_t unpack_wds_GetPacketStatus_t::tXOkBytesCount

8.671.2.9 uint64_t unpack_wds_GetPacketStatus_t::tXOKBytesLastCall

8.671.2.10 uint32_t unpack_wds_GetPacketStatus_t::tXPacketErrors

8.671.2.11 uint32_t unpack_wds_GetPacketStatus_t::tXPacketOverflows

8.671.2.12 uint32_t unpack_wds_GetPacketStatus_t::tXPacketSuccesses

8.672 unpack_wds_GetSessionDuration_t Struct Reference

Data Fields

- uint64_t callDuration

8.672.1 Detailed Description

Parameters

callDuration	call duration in milliseconds

8.672.2 Field Documentation

8.672.2.1 uint64_t unpack_wds_GetSessionDuration_t::callDuration

8.673 unpack_wds_GetSessionState_t Struct Reference

Data Fields

- uint32_t connectionStatus

8.673.1 Detailed Description

Parameters

connection-Status	state of the current packet data session

8.673.2 Field Documentation

8.673.2.1 uint32_t unpack_wds_GetSessionState_t::connectionStatus

8.674 unpack_wds_RMSetTransferStatistics_t Struct Reference

8.675 unpack_wds_SetMobileIPProfile_t Struct Reference

8.676 unpack_wds_SLQSCreateProfile_t Struct Reference

Data Fields

- PackCreateProfileOut * pCreateProfileOut
- uint8_t * pProfileID
- uint16_t Tlvresult

8.676.1 Detailed Description

Parameters

profile	type
profile	index
extended	error

8.676.2 Field Documentation

8.676.2.1 PackCreateProfileOut* unpack_wds_SLQSCreateProfile_t::pCreateProfileOut

8.676.2.2 uint8_t* unpack_wds_SLQSCreateProfile_t::pProfileID

8.676.2.3 uint16_t unpack_wds_SLQSCreateProfile_t::TlvResult

8.677 unpack_wds_SLQSDeleteProfile_t Struct Reference

Data Fields

- uint16_t extendedErrorCode

8.677.1 Detailed Description

Parameters

extendedError-Code	extended error code

8.677.2 Field Documentation

8.677.2.1 uint16_t unpack_wds_SLQSDeleteProfile_t::extendedErrorCode

8.678 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

8.678.1 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

8.678.2 Field Documentation

8.678.2.1 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::_3gppRelease

8.678.2.2 uint8_t unpack_wds_SLQSGet3GPPConfigItem_t::defaultPDNEnabled

8.678.2.3 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfile

8.678.2.4 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileList[24]

8.678.2.5 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::LTEAttachProfileListLen

8.678.2.6 uint16_t unpack_wds_SLQSGet3GPPConfigItem_t::profileList[5]

8.679 unpack_wds_SLQSGetCurrDataSystemStat_t Struct Reference

Data Fields

- uint8_t prefNetwork
- uint8_t networkInfoLen
- currNetworkInfo currNetworkInfo [255]

8.679.1 Detailed Description

Parameters

prefNetwork	preferred network
networkInfoLen	number of set of currNetworkInfo elements
currNetworkInfo	current network infomation.

8.679.2 Field Documentation

8.679.2.1 currNetworkInfo unpack_wds_SLQSGetCurrDataSystemStat_t::currNetworkInfo[255]

8.679.2.2 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::networkInfoLen

8.679.2.3 uint8_t unpack_wds_SLQSGetCurrDataSystemStat_t::prefNetwork

8.680 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

8.680.1 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.

8.680.2 Field Documentation

8.680.2.1 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_rx_rate

8.680.2.2 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::current_channel_tx_rate

8.680.2.3 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_rx_rate

8.680.2.4 uint32_t unpack_wds_SLQSGetCurrentChannelRate_t::max_channel_tx_rate

8.681 unpack_wds_SLQSGetDataBearerTechnology_t Struct Reference

Data Fields

- uint8_t dataBearerMask
- qmiWDSDataBearerTechnology curDataBearerTechnology
- qmiWDSDataBearerTechnology lastCallDataBearerTechnology

8.681.1 Detailed Description

Parameters

dataBearerMask	bit mask indicates bearer info is for current and/or last call
curDataBearer-Technology	current data bearer technology value
lastCallData-Bearer-Technology	last call data bearer technology value

8.681.2 Field Documentation

8.681.2.1 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::curDataBearerTechnology

8.681.2.2 uint8_t unpack_wds_SLQSGetDataBearerTechnology_t::dataBearerMask

8.681.2.3 qmiWDSDataBearerTechnology unpack_wds_SLQSGetDataBearerTechnology_t::lastCallDataBearer-Technology

8.682 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

8.682.1 Detailed Description

Parameters

connection-Status	Connection Status
callEndReason	Last Modem Call End Reason
txOKBytesCount	Tx Bytes OK
rxOKBytesCount	Rx Bytes OK
dormancyStatus	Dormancy Status
dataBearerTech	data bearer technology
channelRate	data Channel Rate
lastCallTXOK-BytesCnt	Last Call Tx Bytes OK
lastCallRXOK-BytesCnt	Last Call Rx Bytes OK
mdmCall-DurationActive	Call active duration
lastCallData-BearerTech	Last Call Data Bearer Technology

8.682.2 Field Documentation

8.682.2.1 uint16_t unpack_wds_SLQSGetDUNCallInfo_t::callEndReason

8.682.2.2 dunchannelRate unpack_wds_SLQSGetDUNCallInfo_t::channelRate

8.682.2.3 connectionStatus unpack_wds_SLQSGetDUNCallInfo_t::connectionStatus

8.682.2.4 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dataBearerTech

8.682.2.5 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::dormancyStatus

8.682.2.6 uint8_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallDataBearerTech

8.682.2.7 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallRXOKBytesCnt

8.682.2.8 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::lastCallTXOKBytesCnt

8.682.2.9 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::mdmCallDurationActive

8.682.2.10 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::rxOKBytesCount

8.682.2.11 uint64_t unpack_wds_SLQSGetDUNCallInfo_t::txOKBytesCount

8.683 unpack_wds_SLQSGetProfileSettings_t Struct Reference

Data Fields

- UnPackGetProfileSettingOut * pProfileSettings
- uint8_t ProfileType
- uint16_t Tlvresult

8.683.1 Field Documentation

8.683.1.1 UnPackGetProfileSettingOut* unpack_wds_SLQSGetProfileSettings_t::pProfileSettings

8.683.1.2 uint8_t unpack_wds_SLQSGetProfileSettings_t::ProfileType

8.683.1.3 uint16_t unpack_wds_SLQSGetProfileSettings_t::Tlvresult

8.684 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

8.684.1 Detailed Description

Parameters

IPv4	ipv4 address
ProfileName	profile name
PDPType	PDP type
APNName	APN name
PrimaryDNSV4	
SecondaryDNS-V4	
UMTSGranted-QoS	UMTS Granted Qos
GPRSGranted-QoS	GPRS Granted QoS
Username	
Authentication	
ProfielID	

GWAddressV4	Gateway IPv4
SubnetMaskV4	Subnet mask IPV4
PCSCFAddrPCO	
PrimaryDNSV6	Primary DNS IPV6
SecondryDNSV6	Secondry DNS IPV6
UMTSGrantedQoS	UMTS Granted Qos
SecondaryDNSV4	
Mtu	Maximum Transfer Unit
DomainList	
IPFamilyPreference	

8.684.2 Field Documentation

8.684.2.1 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::APNName[128]

8.684.2.2 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Authentication

8.684.2.3 struct wds_DomainNameList unpack_wds_SLQSGetRuntimeSettings_t::DomainList

8.684.2.4 struct wds_GPRSQoS unpack_wds_SLQSGetRuntimeSettings_t::GPRSGrantedQoS

8.684.2.5 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::GWAddressV4

8.684.2.6 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IMCNflag

8.684.2.7 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::IPFamilyPreference

8.684.2.8 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::IPv4

8.684.2.9 struct wds_IPV6AddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPV6AddrInfo

8.684.2.10 struct wds_IPV6GWAddressInfo unpack_wds_SLQSGetRuntimeSettings_t::IPV6GWAddrInfo

8.684.2.11 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::Mtu

8.684.2.12 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::PCSCFAddrPCO

8.684.2.13 struct wds_PCSCFFQDNAddressList unpack_wds_SLQSGetRuntimeSettings_t::PCSCFFQDNAddrList

8.684.2.14 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PDPType

8.684.2.15 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV4

8.684.2.16 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::PrimaryDNSV6[8]

8.684.2.17 struct wds_ProfileIdentifier unpack_wds_SLQSGetRuntimeSettings_t::ProfileID

8.684.2.18 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::ProfileName[128]

8.684.2.19 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV4

8.684.2.20 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::SecondaryDNSV6[8]

8.684.2.21 struct wds_PCSCFIPv4ServerAddressList unpack_wds_SLQSGetRuntimeSettings_t::ServerAddrList

8.684.2.22 uint32_t unpack_wds_SLQSGetRuntimeSettings_t::SubnetMaskV4

8.684.2.23 uint16_t unpack_wds_SLQSGetRuntimeSettings_t::Technology

8.684.2.24 LibPackUMTSQoS unpack_wds_SLQSGetRuntimeSettings_t::UMTSGrantedQoS

8.684.2.25 uint8_t unpack_wds_SLQSGetRuntimeSettings_t::Username[128]

8.685 unpack_wds_SLQSModifyProfile_t Struct Reference

Data Fields

- uint16_t * pExtErrorCode

8.685.1 Detailed Description

Parameters

extended	error

8.685.2 Field Documentation

8.685.2.1 uint16_t* unpack_wds_SLQSModifyProfile_t::pExtErrorCode

8.686 unpack_wds_SLQSSetIPFamilyPreference_t Struct Reference

Data Fields

- uint16_t Tlvresult

8.686.1 Detailed Description

Parameters

Tlvresult	unpack result

8.686.2 Field Documentation

8.686.2.1 uint16_t unpack_wds_SLQSSetIPFamilyPreference_t::Tlvresult

8.687 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

8.687.1 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.

8.687.2 Field Documentation

8.687.2.1 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::bearerID

8.687.2.2 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::conn_status

8.687.2.3 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::ipFamily

8.687.2.4 uint8_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::reconfigReqd

8.687.2.5 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::sessionEndReason

8.687.2.6 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::techName

8.687.2.7 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReason

8.687.2.8 uint16_t unpack_wds_SLQSSetPacketSrvStatusCallback_t::verboseSessnEndReasonType

8.688 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

8.688.1 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
dormancyStat-Avail	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.
dBTechExtRat-Value	Data bearer technology Extended RAT Value
dBTechExtSo-Mask	CData bearer technology Extended SO Mask

8.688.2 Field Documentation

8.688.2.1 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::currDBTechAvail

8.688.2.2 wds_currNetworkInfo unpack_wds_SLQSSetWdsEventCallback_ind_t::currNWInfo[255]

8.688.2.3 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dataSysStatAvail

8.688.2.4 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechAvail

8.688.2.5 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechExtAvail

8.688.2.6 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtRatValue

8.688.2.7 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechExtSoMask

8.688.2.8 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBTechnology

8.688.2.9 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dBtechnologyExt

8.688.2.10 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatAvail

8.688.2.11 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::dormancyStatus

8.688.2.12 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipstatAvail

8.688.2.13 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::mipStatus

8.688.2.14 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::netInfoLen

8.688.2.15 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::prefNetwork

8.688.2.16 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::ratMask

8.688.2.17 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_bytes

8.688.2.18 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::rx_pkts

8.688.2.19 uint32_t unpack_wds_SLQSSetWdsEventCallback_ind_t::soMask

8.688.2.20 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_bytes

8.688.2.21 uint64_t unpack_wds_SLQSSetWdsEventCallback_ind_t::tx_pkts

8.688.2.22 uint8_t unpack_wds_SLQSSetWdsEventCallback_ind_t::xferStatAvail

8.689 unpack_wds_SLQSSGetDHCPv4ClientConfig_t Struct Reference

Data Fields

- wdsDhcpv4HwConfig * pHwConfig
- wdsDhcpv4OptionList * pRequestOptionList

8.689.1 Detailed Description

Parameters

pHwConfig	pointer to HW Config structure
pRequestOption-List	pointer to Option List structure to be sent in DHCP request

8.689.2 Field Documentation

8.689.2.1 wdsDhcpv4HwConfig∗ unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pHwConfig

8.689.2.2 wdsDhcpv4OptionList∗ unpack_wds_SLQSSGetDHCPv4ClientConfig_t::pRequestOptionList

8.690 unpack_wds_SLQSSGetLoopback_t Struct Reference

Data Fields

- uint8_t ByteLoopbackMode
- uint8_t ByteLoopbackMultiplier

8.690.1 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.

8.690.2 Field Documentation

8.690.2.1 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMode

8.690.2.2 uint8_t unpack_wds_SLQSSGetLoopback_t::ByteLoopbackMultiplier

8.691 unpack_wds_SLQSStartDataSession_t Struct Reference

Data Fields

- uint32_t ∗ psid
- uint32_t ∗ pFailureReason
- uint32_t ∗ pVerboseFailReasonType
- uint32_t ∗ pVerboseFailureReason

8.691.1 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

8.691.2 Field Documentation

8.691.2.1 uint32_t* unpack_wds_SLQSStartDataSession_t::pFailureReason

8.691.2.2 uint32_t* unpack_wds_SLQSStartDataSession_t::psid

8.691.2.3 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailReasonType

8.691.2.4 uint32_t* unpack_wds_SLQSStartDataSession_t::pVerboseFailureReason

8.692 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]

8.692.1 Detailed Description

Parameters

contextId	Context Identifier
bearerId	Bearer Identity
apnName	APN name associated with the context id
ipv4Address	IPv4 Address
ipv4GWAddress	IPv4 Gateway Address
prDNSIPv4-Address	Primary DNS IPv4 Address
seDNSIPv4-Address	Secondary DNS IPv4 Address
ipv6Address	IPv6 Address

ipv6GWAddress	IPv6 Gateway Address
prDNSIPv6-Address	Primary IPv6 DNS Address
seDNSIPv6-Address	Secondary IPv6 DNS Address
prPCSCFIPv4-Address	Primary PCSCF IPv4 Address
sePCSCFIPv4-Address	Secondary PCSCF IPv4 Address
prPCSCFIPv6-Address	Primary PCSCF IPv6 Address
sePCSCFIPv6-Address	Secondary PCSCF IPv6 Address

8.692.2 Field Documentation

8.692.2.1 int8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::apnName[100]

8.692.2.2 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::bearerId

8.692.2.3 uint8_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::contextId

8.692.2.4 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4Address

8.692.2.5 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv4GWAddress

8.692.2.6 struct ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6Address

8.692.2.7 struct ipv6AddressInfo unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::ipv6GWAddress

8.692.2.8 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv4Address

8.692.2.9 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prDNSIPv6Address[8]

8.692.2.10 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv4Address

8.692.2.11 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::prPCSCFIPv6Address[8]

8.692.2.12 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv4Address

8.692.2.13 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::seDNSIPv6Address[8]

8.692.2.14 uint32_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv4Address

8.692.2.15 uint16_t unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t::sePCSCFIPv6Address[8]

8.693 UnPackGetProfileSettingOut Struct Reference

Data Fields

- UnpackQmiProfileInfo curProfile
- uint16_t * pExtErrCode

8.693.1 Field Documentation

8.693.1.1 UnpackQmiProfileInfo UnPackGetProfileSettingOut::curProfile

8.693.1.2 uint16_t* UnPackGetProfileSettingOut::pExtErrCode

8.694 unpackWdsProfileParam Union Reference

Data Fields

- LibpackProfile3GPP SlqsProfile3GPP
- LibpackProfile3GPP2 SlqsProfile3GPP2

8.694.1 Field Documentation

8.694.1.1 LibpackProfile3GPP unpackWdsProfileParam::SlqsProfile3GPP

8.694.1.2 LibpackProfile3GPP2 unpackWdsProfileParam::SlqsProfile3GPP2

8.695 voice_airTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t airTimerValue

8.695.1 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.

8.695.2 Field Documentation

8.695.2.1 uint32_t voice_airTimer::airTimerValue

8.695.2.2 uint8_t voice_airTimer::namID

8.696 voice_allCallsAlphaIDInfo Struct Reference

Data Fields

- uint8_t callID
- voice_alphaIDInfo AlphaIDInfo

8.696.1 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.

8.696.2 Field Documentation

8.696.2.1 voice_alphaIDInfo voice_allCallsAlphaIDInfo::AlphaIDInfo

8.696.2.2 uint8_t voice_allCallsAlphaIDInfo::callID

8.697 voice_allCallsDiagInfo Struct Reference

Data Fields

- uint8_t callID
- voice_diagInfo DiagInfo

8.697.1 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.

8.697.2 Field Documentation

8.697.2.1 uint8_t voice_allCallsDiagInfo::callID

8.697.2.2 voice_diagInfo voice_allCallsDiagInfo::DiagInfo

8.698 voice_allCallsUUSInfo Struct Reference

Data Fields

- uint8_t callID
- voice_UUSInfo uusInfo

8.698.1 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.

8.698.2 Field Documentation

8.698.2.1 uint8_t voice_allCallsUUSInfo::callID

8.698.2.2 voice_UUSInfo voice_allCallsUUSInfo::uusInfo

8.699 voice_alphaIDInfo Struct Reference

Data Fields

- uint8_t alphaDcs
- uint8_t alphaLen
- uint8_t alphaText [255]

8.699.1 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_DESCRIPTION_LENGTH]	• Data encoded as per the alpha_dcs

8.699.2 Field Documentation

8.699.2.1 uint8_t voice_alphaIDInfo::alphaDcs

8.699.2.2 uint8_t voice_alphaIDInfo::alphaLen

8.699.2.3 uint8_t voice_alphaIDInfo::alphaText[255]

8.700 voice_arrAlertingPattern Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint32_t alertingPattern [20]

8.700.1 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_NO_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

8.700.2 Field Documentation

8.700.2.1 uint32_t voice_arrAlertingPattern::alertingPattern[20]

8.700.2.2 uint8_t voice_arrAlertingPattern::callID[20]

8.700.2.3 uint8_t voice_arrAlertingPattern::numInstances

8.701 voice_arrAlertingType Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint8_t AlertingType [20]

8.701.1 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

8.701.2 Field Documentation

8.701.2.1 uint8_t voice_arrAlertingType::AlertingType[20]

8.701.2.2 uint8_t voice_arrAlertingType::callID[20]

8.701.2.3 uint8_t voice_arrAlertingType::numInstances

8.702 voice_arrAlphaID Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsAlphaIDInfo allCallsAlphaIDInfoArr [20]

8.702.1 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.

8.702.2 Field Documentation

8.702.2.1 voice_allCallsAlphaIDInfo voice_arrAlphaID::allCallsAlphaIDInfoArr[20]

8.702.2.2 uint8_t voice_arrAlphaID::numInstances

8.703 voice_arrCalledPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo CalledPartyNum [20]

8.703.1 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.

8.703.2 Field Documentation

8.703.2.1 voice_peerNumberInfo voice_arrCalledPartyNum::CalledPartyNum[20]

8.703.2.2 uint8_t voice_arrCalledPartyNum::numInstances

8.704 voice_arrCallEndReason Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t callEndReason [20]

8.704.1 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_N-O_OF_CALLS]	• Array of Call End Reason . • See Table9 qaGobiApiTableVoiceCallEndReasons.h for a list of valid voice-related call end reasons

8.704.2 Field Documentation

8.704.2.1 uint16_t voice_arrCallEndReason::callEndReason[20]

8.704.2.2 uint8_t voice_arrCallEndReason::callID[20]

8.704.2.3 uint8_t voice_arrCallEndReason::numInstances

8.705 voice_arrCallInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallInformation getAllCallInfo [20]

8.705.1 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[V-OICE_MAX_NO-_OF_CALLS]	• Array of CallInfo. • See voice_getAllCallInfo for more information.

8.705.2 Field Documentation

8.705.2.1 voice_getAllCallInformation voice_arrCallInfo::getAllCallInfo[20]

8.705.2.2 uint8_t voice_arrCallInfo::numInstances

8.706 voice_arrConnectPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo ConnectedPartyNum [20]

8.706.1 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.

8.706.2 Field Documentation

8.706.2.1 voice_peerNumberInfo voice_arrConnectPartyNum::ConnectedPartyNum[20]

8.706.2.2 uint8_t voice_arrConnectPartyNum::numInstances

8.707 voice_arrDiagInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsDiagInfo DiagInfo [20]

8.707.1 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.

8.707.2 Field Documentation

8.707.2.1 voice_allCallsDiagInfo voice_arrDiagInfo::DiagInfo[20]

8.707.2.2 uint8_t voice_arrDiagInfo::numInstances

8.708 voice_arrRedirPartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_peerNumberInfo RedirPartyNum [20]

8.708.1 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.

8.708.2 Field Documentation

8.708.2.1 uint8_t voice_arrRedirPartyNum::numInstances

8.708.2.2 voice_peerNumberInfo voice_arrRedirPartyNum::RedirPartyNum[20]

8.709 voice_arrRemotePartyName Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyName GetAllCallRmtPtyName [20]

8.709.1 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.

8.709.2 Field Documentation

8.709.2.1 voice_getAllCallRmtPtyName voice_arrRemotePartyName::GetAllCallRmtPtyName[20]

8.709.2.2 uint8_t voice_arrRemotePartyName::numInstances

8.710 voice_arrRemotePartyNum Struct Reference

Data Fields

- uint8_t numInstances
- voice_getAllCallRmtPtyNum RmtPtyNum [20]

8.710.1 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.

8.710.2 Field Documentation

8.710.2.1 uint8_t voice_arrRemotePartyNum::numInstances

8.710.2.2 voice_getAllCallRmtPtyNum voice_arrRemotePartyNum::RmtPtyNum[20]

8.711 voice_arrSvcOption Struct Reference

Data Fields

- uint8_t numInstances
- uint8_t callID [20]
- uint16_t srvOption [20]

8.711.1 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.

8.711.2 Field Documentation

8.711.2.1 uint8_t voice_arrSvcOption::callID[20]

8.711.2.2 uint8_t voice_arrSvcOption::numInstances

8.711.2.3 uint16_t voice_arrSvcOption::srvOption[20]

8.712 voice_arrUUSInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_allCallsUUSInfo AllCallsUUSInfo [20]

8.712.1 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.

8.712.2 Field Documentation

8.712.2.1 voice_allCallsUUSInfo voice_arrUUSInfo::AllCallsUUSInfo[20]

8.712.2.2 uint8_t voice_arrUUSInfo::numInstances

8.713 voice_burstDTMFInfo Struct Reference

Data Fields

- uint8_t * pCallID
- uint8_t digitCnt
- uint8_t pDigitBuff [255]

8.713.1 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

8.713.2 Field Documentation

8.713.2.1 uint8_t voice_burstDTMFInfo::digitCnt

8.713.2.2 uint8_t* voice_burstDTMFInfo::pCallID

8.713.2.3 uint8_t voice_burstDTMFInfo::pDigitBuff[255]

8.714 voice_calledPartySubAdd Struct Reference

Data Fields

- uint8_t extBit
- uint8_t subAddrType
- uint8_t oddEvenInd
- uint8_t subAddrLen
- uint8_t subAddr [255]

8.714.1 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.

8.714.2 Field Documentation

8.714.2.1 uint8_t voice_calledPartySubAdd::extBit

8.714.2.2 uint8_t voice_calledPartySubAdd::oddEvenInd

8.714.2.3 uint8_t voice_calledPartySubAdd::subAddr[255]

8.714.2.4 uint8_t voice_calledPartySubAdd::subAddrLen

8.714.2.5 uint8_t voice_calledPartySubAdd::subAddrType

8.715 voice_callFwdTypeAndPlan Struct Reference

Data Fields

- uint8_t numberType
- uint8_t numberPlan

8.715.1 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

8.715.2 Field Documentation

8.715.2.1 uint8_t voice_callFwdTypeAndPlan::numberPlan

8.715.2.2 uint8_t voice_callFwdTypeAndPlan::numberType

8.716 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]

8.716.1 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.

8.716.2 Field Documentation

8.716.2.1 uint8_t voice_callFWExtInfo::noReplyTimer

8.716.2.2 uint8_t voice_callFWExtInfo::number[255]

8.716.2.3 uint8_t voice_callFWExtInfo::numLen

8.716.2.4 uint8_t voice_callFWExtInfo::numPlan

8.716.2.5 uint8_t voice_callFWExtInfo::numType

8.716.2.6 uint8_t voice_callFWExtInfo::PI

8.716.2.7 uint8_t voice_callFWExtInfo::SI

8.716.2.8 uint8_t voice_callFWExtInfo::SvcClass

8.716.2.9 uint8_t voice_callFWExtInfo::SvcStatus

8.717 voice_callFWInfo Struct Reference

Data Fields

- uint8_t SvcStatus
- uint8_t SvcClass
- uint8_t numLen
- uint8_t number [255]
- uint8_t noReplyTimer

8.717.1 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.

8.717.2 Field Documentation

8.717.2.1 uint8_t voice_callFWInfo::noReplyTimer

8.717.2.2 uint8_t voice_callFWInfo::number[255]

8.717.2.3 uint8_t voice_callFWInfo::numLen

8.717.2.4 uint8_t voice_callFWInfo::SvcClass

8.717.2.5 uint8_t voice_callFWInfo::SvcStatus

8.718 voice_callInfo Struct Reference

Data Fields

- uint8_t callID
- uint8_t callState
- uint8_t callType
- uint8_t direction
- uint8_t mode

8.718.1 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

8.718.2 Field Documentation

8.718.2.1 uint8_t voice_callInfo::callID

8.718.2.2 uint8_t voice_callInfo::callState

8.718.2.3 uint8_t voice_callInfo::callType

8.718.2.4 uint8_t voice_callInfo::direction

8.718.2.5 uint8_t voice_callInfo::mode

8.719 voice_ccSUPSType Struct Reference

Data Fields

- uint8_t svcType
- uint8_t reason

8.719.1 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.

8.719.2 Field Documentation

8.719.2.1 uint8_t voice_ccSUPSType::reason

8.719.2.2 uint8_t voice_ccSUPSType::svcType

8.720 voice_CLIPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.720.1 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

8.720.2 Field Documentation

8.720.2.1 uint8_t voice_CLIPResp::ActiveStatus

8.720.2.2 uint8_t voice_CLIPResp::ProvisionStatus

8.721 voice_CLIRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.721.1 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

8.721.2 Field Documentation

8.721.2.1 uint8_t voice_CLIRResp::ActiveStatus

8.721.2.2 uint8_t voice_CLIRResp::ProvisionStatus

8.722 voice_CNAPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.722.1 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

8.722.2 Field Documentation

8.722.2.1 uint8_t voice_CNAPResp::ActiveStatus

8.722.2.2 uint8_t voice_CNAPResp::ProvisionStatus

8.723 voice_COLPResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.723.1 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

8.723.2 Field Documentation

8.723.2.1 uint8_t voice_COLPResp::ActiveStatus

8.723.2.2 uint8_t voice_COLPResp::ProvisionStatus

8.724 voice_COLRResp Struct Reference

Data Fields

- uint8_t ActiveStatus
- uint8_t ProvisionStatus

8.724.1 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

8.724.2 Field Documentation

8.724.2.1 uint8_t voice_COLRResp::ActiveStatus

8.724.2.2 uint8_t voice_COLRResp::ProvisionStatus

8.725 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]

8.725.1 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.

8.725.2 Field Documentation

8.725.2.1 uint8_t voice_connectNumInfo::callerID[81]

8.725.2.2 uint8_t voice_connectNumInfo::callerIDLen

8.725.2.3 uint8_t voice_connectNumInfo::numPlan

8.725.2.4 uint8_t voice_connectNumInfo::numPresInd

8.725.2.5 uint8_t voice_connectNumInfo::numType

8.725.2.6 uint8_t voice_connectNumInfo::screeningInd

8.726 voice_CUGInfo Struct Reference

Data Fields

- uint16_t CUGIndex
- uint8_t SuppPrefCUG
- uint8_t SuppOA

8.726.1 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

8.726.2 Field Documentation

8.726.2.1 uint16_t voice_CUGInfo::CUGIndex

8.726.2.2 uint8_t voice_CUGInfo::SuppOA

8.726.2.3 uint8_t voice_CUGInfo::SuppPrefCUG

8.727 voice_curAMRConfig Struct Reference

Data Fields

- uint8_t gsmAmrStat
- uint8_t wcdmaAmrStat

8.727.1 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

8.727.2 Field Documentation

8.727.2.1 uint8_t voice_curAMRConfig::gsmAmrStat

8.727.2.2 uint8_t voice_curAMRConfig::wcdmaAmrStat

8.728 voice_diagInfo Struct Reference

Data Fields

- uint8_t diagInfoLen
- uint8_t diagnosticInfo [255]

8.728.1 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.

8.728.2 Field Documentation

8.728.2.1 uint8_t voice_diagInfo::diagInfoLen

8.728.2.2 uint8_t voice_diagInfo::diagnosticInfo[255]

8.729 voice_DTMFLengths Struct Reference

Data Fields

- uint8_t DTMFPulseWidth
- uint8_t DTMFInterdigitInterval

8.729.1 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

8.729.2 Field Documentation

8.729.2.1 uint8_t voice_DTMFLengths::DTMFInterdigitInterval

8.729.2.2 uint8_t voice_DTMFLengths::DTMFPulseWidth

8.730 voice_getAllCallInformation Struct Reference

Data Fields

- voice_callInfo CallInfo
- uint8_t isEmpty
- uint8_t ALS

8.730.1 Detailed Description

This structure contains information related to call state change.

Parameters

CallInfo	• See 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

8.730.2 Field Documentation

8.730.2.1 uint8_t voice_getAllCallInformation::ALS

8.730.2.2 voice_callInfo voice_getAllCallInformation::CallInfo

8.730.2.3 uint8_t voice_getAllCallInformation::isEmpty

8.731 voice_getAllCallRmtPtyName Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyName RemotePartyName

8.731.1 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.

8.731.2 Field Documentation

8.731.2.1 uint8_t voice_getAllCallRmtPtyName::callID

8.731.2.2 voice_remotePartyName voice_getAllCallRmtPtyName::RemotePartyName

8.732 voice_getAllCallRmtPtyNum Struct Reference

Data Fields

- uint8_t callID
- voice_remotePartyNum RemotePartyNum

8.732.1 Detailed Description

This structure contains information for All Call Remote Party Numbers

Parameters

callID	• Unique call identifier for the call.
RemoteParty-Num	• See remotePartyNum for more information.

8.732.2 Field Documentation

8.732.2.1 uint8_t voice_getAllCallRmtPtyNum::callID

8.732.2.2 voice_remotePartyNum voice_getAllCallRmtPtyNum::RemotePartyNum

8.733 voice_getCallFWExtInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWExtInfo CallFWExtInfo [20]

8.733.1 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_NO_OF_CALLS]	• Array of CallFWExtInfo. – See voice_CallFWExtInfo for more information.

8.733.2 Field Documentation

8.733.2.1 voice_callFWExtInfo voice_getCallFWExtInfo::CallFWExtInfo[20]

8.733.2.2 uint8_t voice_getCallFWExtInfo::numInstances

8.734 voice_getCallFWInfo Struct Reference

Data Fields

- uint8_t numInstances
- voice_callFWInfo CallFWInfo [20]

8.734.1 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.

8.734.2 Field Documentation

8.734.2.1 voice_callFWInfo voice_getCallFWInfo::callFWInfo[20]

8.734.2.2 uint8_t voice_getCallFWInfo::numInstances

8.735 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]

8.735.1 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.

8.735.2 Field Documentation

8.735.2.1 uint8_t voice_peerNumberInfo::callID

8.735.2.2 uint8_t voice_peerNumberInfo::number[81]

8.735.2.3 uint8_t voice_peerNumberInfo::numLen

8.735.2.4 uint8_t voice_peerNumberInfo::numPI

8.735.2.5 uint8_t voice_peerNumberInfo::numPlan

8.735.2.6 uint8_t voice_peerNumberInfo::numSI

8.735.2.7 uint8_t voice_peerNumberInfo::numType

8.736 voice_prefVoiceSO Struct Reference

Data Fields

- uint8_t namID

- uint8_t evrcCapability
- uint16_t homePageVoiceSO
- uint16_t homeOrigVoiceSO
- uint16_t roamOrigVoiceSO

8.736.1 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

8.736.2 Field Documentation

8.736.2.1 uint8_t voice_prefVoiceSO::evrcCapability

8.736.2.2 uint16_t voice_prefVoiceSO::homeOrigVoiceSO

8.736.2.3 uint16_t voice_prefVoiceSO::homePageVoiceSO

8.736.2.4 uint8_t voice_prefVoiceSO::namID

8.736.2.5 uint16_t voice_prefVoiceSO::roamOrigVoiceSO

8.737 voice_remotePartyName Struct Reference

Data Fields

- uint8_t namePI
- uint8_t codingScheme

- uint8_t nameLen
- uint8_t callerName [255]

8.737.1 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.

8.737.2 Field Documentation

8.737.2.1 uint8_t voice_remotePartyName::callerName[255]

8.737.2.2 uint8_t voice_remotePartyName::codingScheme

8.737.2.3 uint8_t voice_remotePartyName::nameLen

8.737.2.4 uint8_t voice_remotePartyName::namePI

8.738 voice_remotePartyNum Struct Reference

Data Fields

- uint8_t presentationInd
- uint8_t numLen
- uint8_t remPartyNumber [81]

8.738.1 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.

8.738.2 Field Documentation

8.738.2.1 uint8_t voice_remotePartyNum::numLen

8.738.2.2 uint8_t voice_remotePartyNum::presentationInd

8.738.2.3 uint8_t voice_remotePartyNum::remPartyNumber[81]

8.739 voice_roamTimer Struct Reference

Data Fields

- uint8_t namID
- uint32_t roamTimerValue

8.739.1 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.

8.739.2 Field Documentation

8.739.2.1 uint8_t voice_roamTimer::namID

8.739.2.2 uint32_t voice_roamTimer::roamTimerValue

8.740 voice_USSInfo Struct Reference

Data Fields

- uint8_t ussDCS
- uint8_t ussLen
- uint8_t ussData [182]

8.740.1 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

8.740.2 Field Documentation

8.740.2.1 uint8_t voice_USSInfo::ussData[182]

8.740.2.2 uint8_t voice_USSInfo::ussDCS

8.740.2.3 uint8_t voice_USSInfo::ussLen

8.741 voice_UUSInfo Struct Reference

Data Fields

- uint8_t UUSType

- uint8_t UUSDcs
- uint8_t UUSDatalen
- uint8_t UUSData [255]

8.741.1 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

8.741.2 Field Documentation

8.741.2.1 uint8_t voice_UUSInfo::UUSData[255]

8.741.2.2 uint8_t voice_UUSInfo::UUSDatalen

8.741.2.3 uint8_t voice_UUSInfo::UUSDcs

8.741.2.4 uint8_t voice_UUSInfo::UUSType

8.742 wds_currNetworkInfo Struct Reference

Data Fields

- uint8_t NetworkType
- uint32_t RATMask
- uint32_t SOMask

8.742.1 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

8.742.2 Field Documentation

8.742.2.1 uint8_t wds_currNetworkInfo::NetworkType

8.742.2.2 uint32_t wds_currNetworkInfo::RATMask

8.742.2.3 uint32_t wds_currNetworkInfo::SOMask

8.743 wds_DataULongLongTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint64_t ullData

8.743.1 Field Documentation

8.743.1.1 uint8_t wds_DataULongLongTlv::TlvPresent

8.743.1.2 uint64_t wds_DataULongLongTlv::ullData

8.744 wds_DataULongTlv Struct Reference

Data Fields

- uint8_t TlvPresent

- uint32_t ulData

8.744.1 Field Documentation

8.744.1.1 uint8_t wds_DataULongTlv::TlvPresent

8.744.1.2 uint32_t wds_DataULongTlv::ulData

8.745 wds_DHCPLeaseOptTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t numOpt
- wds_DHCPOpt optList [30]
- uint8_t optListData [2048]

8.745.1 Field Documentation

8.745.1.1 uint8_t wds_DHCPLeaseOptTlv::numOpt

8.745.1.2 wds_DHCPOpt wds_DHCPLeaseOptTlv::optList[30]

8.745.1.3 uint8_t wds_DHCPLeaseOptTlv::optListData[2048]

8.745.1.4 uint8_t wds_DHCPLeaseOptTlv::TlvPresent

8.746 wds_DHCPLeaseStateTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t leaseState

8.746.1 Field Documentation

8.746.1.1 uint8_t wds_DHCPLeaseStateTlv::leaseState

8.746.1.2 uint8_t wds_DHCPLeaseStateTlv::TlvPresent

8.747 wds_DHCPOpt Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t * pOptVal

8.747.1 Field Documentation

8.747.1.1 uint8_t wds_DHCPOpt::optCode

8.747.1.2 uint8_t wds_DHCPOpt::optValLen

8.747.1.3 uint8_t* wds_DHCPOpt::pOptVal

8.748 wds_DHCPProfileIdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint8_t profileType
- uint8_t profileId

8.748.1 Field Documentation

8.748.1.1 uint8_t wds_DHCPProfileIdTlv::profileId

8.748.1.2 uint8_t wds_DHCPProfileIdTlv::profileType

8.748.1.3 uint8_t wds_DHCPProfileIdTlv::TlvPresent

8.749 wds_DHCPv4HWConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.749.1 Detailed Description

WDS SWI DHCPv4 HW Config Structure.

Parameters

hwType	• HW Type 1 - Ethernet 20 - Serial
chaddrlen	• chaddrlen
chaddr	• chaddr. Max size 16 bytes

8.749.2 Field Documentation

8.749.2.1 uint8_t wds_DHCPv4HWConfig::chaddr[16]

8.749.2.2 uint8_t wds_DHCPv4HWConfig::chaddrLen

8.749.2.3 uint8_t wds_DHCPv4HWConfig::hwType

8.750 wds_DHCPv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.750.1 Detailed Description

WDS SWI DHCPv4 Option Structure

Parameters

optCode	• Option code – 0 - 255
optValLen	• Option value length – 0 - 255
optVal	• Option value

8.750.2 Field Documentation

8.750.2.1 uint8_t wds_DHCPv4Option::optCode

8.750.2.2 uint8_t wds_DHCPv4Option::optVal[255]

8.750.2.3 uint8_t wds_DHCPv4Option::optValLen

8.751 wds_DHCPv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wds_DHCPv4Option * pOptList

8.751.1 Detailed Description

WDS SWI DHCPv4 Option List Structure

Parameters

numOpt	• number of options – 0 - 255
pOptList	• pointer to list of DHCP Options

8.751.2 Field Documentation

8.751.2.1 uint8_t wds_DHCPv4OptionList::numOpt

8.751.2.2 wds_DHCPv4Option* wds_DHCPv4OptionList::pOptList

8.752 wds_DHCPv4ProfileId Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileId

8.752.1 Detailed Description

WDS SWI DHCPv4 Profile Identifier Structure

Parameters

profileType	• 0 for 3GPP
profileId	• 1 to 24 for 3GPP profile

8.752.2 Field Documentation

8.752.2.1 uint8_t wds_DHCPv4ProfileId::profileId

8.752.2.2 uint8_t wds_DHCPv4ProfileId::profileType

8.753 wds_Domain Struct Reference

Data Fields

- uint16_t domainLen
- uint8_t domainName [256]

8.753.1 Detailed Description

This structure contains the DomainName Information

Parameters

domainLen	• length of the recieved Domain name
domainName	• Domain name(Max 256 characters)

8.753.2 Field Documentation

8.753.2.1 uint16_t wds_Domain::domainLen

8.753.2.2 uint8_t wds_Domain::domainName[256]

8.754 wds_DomainNameList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_Domain domain [10]

8.754.1 Detailed Description

This structure contains the DomainNameList Information

Parameters

numInstances	• Number of Domain name receieved
domain	• Domain name information(Max 10 Domain names)

8.754.2 Field Documentation

8.754.2.1 struct wds_Domain wds_DomainNameList::domain[10]

8.754.2.2 uint8_t wds_DomainNameList::numInstances

8.755 wds_GPRSQoS Struct Reference

Data Fields

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

8.755.1 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

8.755.2 Field Documentation

8.755.2.1 uint32_t wds_GPRSQoS::delayClass

8.755.2.2 uint32_t wds_GPRSQoS::meanThroughputClass

8.755.2.3 uint32_t wds_GPRSQoS::peakThroughputClass

8.755.2.4 uint32_t wds_GPRSQoS::precedenceClass

8.755.2.5 uint32_t wds_GPRSQoS::reliabilityClass

8.756 wds_IPv4AdTlv Struct Reference

Data Fields

- uint8_t TlvPresent
- uint32_t IPv4Addr

8.756.1 Field Documentation

8.756.1.1 uint32_t wds_IPv4AdTlv::IPv4Addr

8.756.1.2 uint8_t wds_IPv4AdTlv::TlvPresent

8.757 wds_IPV6AddressInfo Struct Reference

Data Fields

- uint8_t IPV6PrefixLen
- uint16_t IPAddressV6 [8]

8.757.1 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.

8.757.2 Field Documentation

8.757.2.1 uint16_t wds_IPV6AddressInfo::IPAddressV6[8]

8.757.2.2 uint8_t wds_IPV6AddressInfo::IPV6PrefixLen

8.758 wds_IPV6GWAddressInfo Struct Reference

Data Fields

- uint8_t gwV6PrefixLen
- uint16_t gwAddressV6 [8]

8.758.1 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.

8.758.2 Field Documentation

8.758.2.1 uint16_t wds_IPV6GWAddressInfo::gwAddressV6[8]

8.758.2.2 uint8_t wds_IPV6GWAddressInfo::gwV6PrefixLen

8.759 wds_PCSCFFQDNAddress Struct Reference

Data Fields

- uint16_t fqdnLen

- uint8_t fqdnAddr [256]

8.759.1 Detailed Description

This structure contains the PCSCFFQDNAddress Information

Parameters

fqdnLen	• length of the received FQDN address
fqdnAddr	• FQDN address(Max 256 characters)

8.759.2 Field Documentation

8.759.2.1 uint8_t wds_PCSCFFQDNAddress::fqdnAddr[256]

8.759.2.2 uint16_t wds_PCSCFFQDNAddress::fqdnLen

8.760 wds_PCSCFFQDNAddressList Struct Reference

Data Fields

- uint8_t numInstances
- struct wds_PCSCFFQDNAddress pcsfFQDNAddress [10]

8.760.1 Detailed Description

This structure contains the PCSCFFQDNAddressList Information

Parameters

numInstances	• Number of FQDN addresses received
pcsfFQDN-Address	• FQDN address information(Max 10 addresses)

8.760.2 Field Documentation

8.760.2.1 uint8_t wds_PCSCFFQDNAddressList::numInstances

8.760.2.2 struct wds_PCSCFFQDNAddress wds_PCSCFFQDNAddressList::pcsfFQDNAddress[10]

8.761 wds_PCSCFIPv4ServerAddressList Struct Reference

Data Fields

- uint8_t numInstances
- uint32_t pscsfIPv4Addr [64]

8.761.1 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)

8.761.2 Field Documentation

8.761.2.1 uint8_t wds_PCSCFIPv4ServerAddressList::numInstances

8.761.2.2 uint32_t wds_PCSCFIPv4ServerAddressList::pscsfIPv4Addr[64]

8.762 wds_ProfileIdentifier Struct Reference

Data Fields

- uint8_t profileType
- uint8_t profileIndex

8.762.1 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

8.762.2 Field Documentation

8.762.2.1 uint8_t wds_ProfileIdentifier::profileIndex

8.762.2.2 uint8_t wds_ProfileIdentifier::profileType

8.763 wds_profileInfo Union Reference

Data Fields

- LibPackprofile_3GPP SlqsProfile3GPP
- LibPackprofile_3GPP2 SlqsProfile3GPP2

8.763.1 Detailed Description

This union consist of profile_3GPP and profile_3GPP2 out of which one will be used to create profile.

8.763.2 Field Documentation

8.763.2.1 LibPackprofile_3GPP wds_profileInfo::SlqsProfile3GPP

8.763.2.2 LibPackprofile_3GPP2 wds_profileInfo::SlqsProfile3GPP2

8.764 wds_TrStatInd Struct Reference

Data Fields

- uint8_t statsPeriod
- uint32_t statsMask

8.764.1 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.

8.764.2 Field Documentation

8.764.2.1 uint32_t wds_TrStatInd::statsMask

8.764.2.2 uint8_t wds_TrStatInd::statsPeriod

8.765 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

8.765.1 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.

8.765.2 Field Documentation

8.765.2.1 uint8_t wds_UMTSMinQoS::deliveryErrSDU

8.765.2.2 uint32_t wds_UMTSMinQoS::grntDownlinkBitrate

8.765.2.3 uint32_t wds_UMTSMinQoS::grntUplinkBitrate

8.765.2.4 uint32_t wds_UMTSMinQoS::maxDownlinkBitrate

8.765.2.5 uint32_t wds_UMTSMinQoS::maxSDUSize

8.765.2.6 uint32_t wds_UMTSMinQoS::maxUplinkBitrate

8.765.2.7 uint8_t wds_UMTSMinQoS::qosDeliveryOrder

8.765.2.8 uint8_t wds_UMTSMinQoS::resBerRatio

8.765.2.9 uint8_t wds_UMTSMinQoS::sduErrorRatio

8.765.2.10 uint8_t wds_UMTSMinQoS::trafficClass

8.765.2.11 uint32_t wds_UMTSMinQoS::trafficPriority

8.765.2.12 uint32_t wds_UMTSMinQoS::transferDelay

8.766 wdsDhcpv4HwConfig Struct Reference

Data Fields

- uint8_t hwType
- uint8_t chaddrLen
- uint8_t chaddr [16]

8.766.1 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

8.766.2 Field Documentation

8.766.2.1 uint8_t wdsDhcpv4HwConfig::chaddr[16]

8.766.2.2 uint8_t wdsDhcpv4HwConfig::chaddrLen

8.766.2.3 uint8_t wdsDhcpv4HwConfig::hwType

8.767 wdsDhcpv4Option Struct Reference

Data Fields

- uint8_t optCode
- uint8_t optValLen
- uint8_t optVal [255]

8.767.1 Detailed Description

Parameters

optCode	Option code • 0 - 255
optValLen	Option value length • 0 - 255
optVal	Option Value

8.767.2 Field Documentation

8.767.2.1 uint8_t wdsDhcpv4Option::optCode

8.767.2.2 uint8_t wdsDhcpv4Option::optVal[255]

8.767.2.3 uint8_t wdsDhcpv4Option::optValLen

8.768 wdsDhcpv4OptionList Struct Reference

Data Fields

- uint8_t numOpt
- wdsDhcpv4Option * pOptList

8.768.1 Detailed Description

Parameters

numOpt	number of options • 0 - 255
pOptList	pointer to list of DHCP Options

8.768.2 Field Documentation

8.768.2.1 uint8_t wdsDhcpv4OptionList::numOpt

8.768.2.2 wdsDhcpv4Option* wdsDhcpv4OptionList::pOptList

8.769 wdsDhcpv4ProfileId Struct Reference

Data Fields

- uint8_t profileType

- uint8_t profileId

8.769.1 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)

8.769.2 Field Documentation

8.769.2.1 uint8_t wdsDhcpv4ProfileId::profileId

8.769.2.2 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 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,
  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 ()

Variables

- logger glog
- uint8_t gloglvl

9.2.1 Macro Definition Documentation

9.2.1.1 #define DEAULT_LOC_TIMEOUT_IN_SEC 2

9.2.1.2 #define MINREQBKLEN (2048)

9.2.1.3 #define MSGID_AND_LEN (4)

9.2.1.4 #define MSGID_DONT_CARE (0xffff)

9.2.1.5 #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; \
        }
```

9.2.1.6 #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;
```

9.2.1.7 #define SDU_HDR_LEN (3)

9.2.1.8 #define UNUSEDPARAM(*x*) (void)x

9.2.2 Typedef Documentation

9.2.2.1 typedef void(∗ logger)(uint8_t lvl, const char ∗buff)

9.2.3 Enumeration Type Documentation

9.2.3.1 enum eLOG_LEVEL

log levels

Enumerator

eLOG_INFO

eLOG_DEBUG

eLOG_WARN

eLOG_FATAL

9.2.3.2 enum eQMI_SVC

qmi service

Enumerator

eCTL

eWDS

eDMS

eNAS

eQOS

eSMS

ePDS

eVOICE

eCAT

eUIM

eLOC

eSAR

eTMD

eRMS

eSWIOMA

eSWIM2MCMD

eSWIOMAEXT

eSWILOC

9.2.3.3 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

9.2.3.4 enum msgtype

qmi message type

Enumerator

eREQ

eRSP

eIND

9.2.4 Function Documentation

9.2.4.1 void fill_pack_ctx (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* uint8_t *svc,* int *timeout*)

9.2.4.2 void fill_sdu_hdr (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf*)

9.2.4.3 char* get_version ( )

Returns

version string

9.2.4.4 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

9.2.4.5 uint16_t helper_get_xid (uint8_t * *qmi_resp*)

9.2.4.6 int helper_set_log_func (logger *func*)

set log function

9.2.4.7 int helper_set_log_lvl (uint8_t *lvl*)

set log level

9.2.4.8 char* liteqmi_GetVersion ( )

Returns

version string

9.2.4.9 void liteqmi_log (uint8_t *lvl*, const char * *fmt*, ...)

9.2.4.10 unsigned unpack_result_code_only (uint8_t * *pMdmResp*)

common handler for unpacking response with TLV type 0x02 only

9.2.5 Variable Documentation

9.2.5.1 logger glog

9.2.5.2 uint8_t gloglvl

9.3 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 ∗p-Output)
- 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)

9.3.1 Macro Definition Documentation

9.3.1.1 #define ACT_CODE_MAX_SIZE 81

9.3.1.2 #define CK_MAX_SIZE 8

9.3.1.3 #define DMS_IMGDETAILS_LEN 16

9.3.1.4 #define DMS_MAX_CUST_ID_LEN 64

9.3.1.5 #define DMS_MAX_CUST_VALUE_LEN 8

9.3.1.6 #define DMS_MAX_FWUPDATE_LOG_STR_SZ 255

9.3.1.7 #define DMS_MAX_FWUPDATE_REF_STR_SZ 15

9.3.1.8 #define DMS_PM_FACTORY 0x02 /* Factory Test Mode */

9.3.1.9 #define DMS_PM_LOW 0x01 /* Low Power */

9.3.1.10 #define DMS_PM_OFFLINE 0x03 /* Offline */

9.3.1.11 #define DMS_PM_ONLINE 0x00 /* Online */

9.3.1.12 #define DMS_PM_PERSISTENT_LOW 0x06 /* Persistent Low Power */

9.3.1.13 #define DMS_PM_RESET 0x04 /* Reset */

9.3.1.14 #define DMS_PM_SHUT_DOWN 0x05 /* Shut Down */

9.3.1.15 #define DMS_SET_REPORT_DISABLE 0

9.3.1.16 #define DMS_SET_REPORT_ENABLE 1

9.3.1.17 #define DMS_SLQSFWINFO_APPVERSION_SZ 85

9.3.1.18 #define DMS_SLQSFWINFO_BOOTVERSION_SZ 85

9.3.1.19 #define DMS_SLQSFWINFO_CARRIER_SZ 20

9.3.1.20 #define DMS_SLQSFWINFO_CUR_CARR_NAME 17

9.3.1.21 #define DMS_SLQSFWINFO_CUR_CARR_REV 13

9.3.1.22 #define DMS_SLQSFWINFO_MODELID_SZ 20

9.3.1.23 #define DMS_SLQSFWINFO_PACKAGEID_SZ 85

9.3.1.24 #define DMS_SLQSFWINFO_PRIVERSION_SZ 16

9.3.1.25 #define DMS_SLQSFWINFO_SKU_SZ 15

9.3.1.26 #define DMS_SWI_SET_IND_DISABLE 0

9.3.1.27 #define DMS_SWI_SET_IND_ENABLE 1

9.3.1.28 #define DMS_UINT8_MAX_STRING_SZ 255

9.3.1.29 #define ERI_DATA_MAX_SIZE 1024

9.3.1.30 #define MAX_BUILD_ID_LEN 255

9.3.1.31 #define MEID_MAX_SIZE 8

9.3.1.32 #define SLQS_MAX_DYING_GASP_CFG_SMS_CONTENT_LENGTH 160

9.3.1.33 #define SLQS_MAX_DYING_GASP_CFG_SMS_NUMBER_LENGTH 20

9.3.1.34 #define SPC_SIZE 6

9.3.1.35 #define UNIQUE_ID_LEN 16

9.3.2 Function Documentation

9.3.2.1 int pack_dms_ActivateAutomatic (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ActivateAutomatic_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

9.3.2.2 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

9.3.2.3 int pack_dms_GetBandCapability (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Band Capability pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.4 int pack_dms_GetCrashAction (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Crash Action pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.5 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.6 int pack_dms_GetCustFeaturesV2 (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_GetCustFeaturesV2_t * *reqArg*)

9.3.2.7 int pack_dms_GetDeviceCap (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Device Capabilities pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.8 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.9 int pack_dms_GetDeviceHardwareRev (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Hardware Revision pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.10 int pack_dms_GetDeviceMfr (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Manufacture pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.11 int pack_dms_GetDeviceSerialNumbers (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

get Device Serial Number pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.12 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.13 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.14 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.15 int pack_dms_GetFSN (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get FSN pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.16 int pack_dms_GetHardwareRevision (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

get hardware revision 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

9.3.2.17 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.18 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

9.3.2.19 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.20 int pack_dms_GetNetworkTime (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Network Time pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.21 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

9.3.2.22 int pack_dms_GetPower (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

get power 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

9.3.2.23 int pack_dms_GetPRLVersion (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get PRL Versions pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.24 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.25 int pack_dms_GetUSBComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get USB Comp pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.26 int pack_dms_GetVoiceNumber (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Get Voice Number pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.27 int pack_dms_ResetToFactoryDefaults (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ResetToFactoryDefaults_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

9.3.2.28 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.29 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.30 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.31 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.32 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.33 int pack_dms_SetFirmwarePreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, void * *reqArg*)

Set Firmware Preference pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.34 int pack_dms_SetPower (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetPower_t * *reqArg*)

Set Power pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.35 int pack_dms_SetUSBComp (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SetUSBComp_t * *reqArg*)

Set USB Comp pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.36 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

9.3.2.37 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

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

9.3.2.38 int pack_dms_SLQSGetBandCapability (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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.39 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
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

9.3.2.40 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

9.3.2.41 int pack_dms_SLQSSwiGetCrashInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_SLQSSwiGetCrashInfo_t * *pReq*)

To set Get Crash 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

9.3.2.42 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

9.3.2.43 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

9.3.2.44 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

9.3.2.45 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

9.3.2.46 int pack_dms_SLQSSwiGetHostDevInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set Host Dev 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

9.3.2.47 int pack_dms_SLQSSwiGetOSInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set 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

9.3.2.48 int pack_dms_SLQSSwiGetSerialNoExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set 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

9.3.2.49 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.3.2.50 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

9.3.2.51 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

9.3.2.52 int pack_dms_SLQSUIMGetState (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

To set UIM Get State 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

9.3.2.53 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

9.3.2.54 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

9.3.2.55 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.3.2.56 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

9.3.2.57 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

9.3.2.58 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

9.3.2.59 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

9.3.2.60 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

9.3.2.61 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

9.3.2.62 int pack_dms_ValidateSPC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_dms_ResetToFactoryDefaults_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

9.3.2.63 int unpack_dms_ActivateAutomatic (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactoryDefaults_t * *pOutput*)

To get UIM Unblock Control Key unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.64 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	*pespLen*	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

9.3.2.65 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	*pespLen*	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

9.3.2.66 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	*pespLen*	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

9.3.2.67 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	*pespLen*	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

9.3.2.68 int unpack_dms_GetCustFeaturesV2 (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetCustFeaturesV2_t * *pOutput*)

9.3.2.69 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	*pespLen*	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

9.3.2.70 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	*pespLen*	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

9.3.2.71 int unpack_dms_GetDeviceHardwareRev (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceHardwareRev_t * *pOutput*)

Get Hardware Revision unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.72 int unpack_dms_GetDeviceMfr (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetDeviceMfr_t * *pOutput*)

Get Manufacture unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.73 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	*pespLen*	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

9.3.2.74 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	*pespLen*	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

9.3.2.75 int unpack_dms_GetFirmwareRevision (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmware-Revision_t * *pOutput*)

get Firmware Revision unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.76 int unpack_dms_GetFirmwareRevisions (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFirmware-Revisions_t * *pOutput*)

get Firmware Revisions unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.77 int unpack_dms_GetFSN (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetFSN_t * *pOutput*)

Get FSN unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.78 int unpack_dms_GetHardwareRevision (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetHardwareRevision_t * *pOutput*)

get hardware revision unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.79 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	*pespLen*	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

9.3.2.80 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	*pespLen*	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

9.3.2.81 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	*pespLen*	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

9.3.2.82 int unpack_dms_GetNetworkTime (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetNetworkTime_t * *pOutput*)

Get Network Time unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.83 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	*pespLen*	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

9.3.2.84 int unpack_dms_GetPower (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetPower_t * *pOutput*)

get power unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.85 int unpack_dms_GetPRLVersion (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetPRLVersion_t * *pOutput*)

Get PRL Versions unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.86 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	*pespLen*	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

9.3.2.87 int unpack_dms_GetUSBComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetUSBComp_t * *pOutput*)

Get USB Comp unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.88 int unpack_dms_GetVoiceNumber (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_GetVoiceNumber_t * *pOutput*)

Get Voice Number unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.89 int unpack_dms_ResetToFactoryDefaults (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactoryDefaults_t * *pOutput*)

To get UIM Unblock Control Key unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.90 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	*pespLen*	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

9.3.2.91 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	*pespLen*	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

9.3.2.92 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	*pespLen*	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

9.3.2.93 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	*pespLen*	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

9.3.2.94 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	*pespLen*	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

9.3.2.95 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	*pespLen*	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

9.3.2.96 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	*pespLen*	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

9.3.2.97 int unpack_dms_SetPower (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetPower_t * *pOutput*)

Set Power unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.98 int unpack_dms_SetUSBComp (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SetUSBComp_t * *pOutput*)

Set USB Comp unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.99 int unpack_dms_SLQSDmsSwiGetResetInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSDmsSwiGetResetInfo_t * *pOutput*)

To get reset info unpack

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

9.3.2.100 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	*pespLen*	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

9.3.2.101 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	*pespLen*	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

9.3.2.102 int unpack_dms_SLQSGetBandCapability (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSGetBandCapability_t * *pOutput*)

get band capability unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.103 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	*pespLen*	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

9.3.2.104 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	*pespLen*	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

9.3.2.105 int unpack_dms_SLQSSwiGetCrashInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetCrashInfo_t * *pOutput*)

To get Crash Info unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.106 int unpack_dms_SLQSSwiGetDyingGaspCfg (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetDyingGaspCfg_t * *pOutput*)

Get Dying GASP Config unpack

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

9.3.2.107 int unpack_dms_SLQSSwiGetDyingGaspStatistics (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetDyingGaspStatistics_t * *pOutput*)

Get Dying GASP Statistics unpack

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

9.3.2.108 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	*pespLen*	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

9.3.2.109 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	*pespLen*	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

9.3.2.110 int unpack_dms_SLQSSwiGetHostDevInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiGetHostDevInfo_t * *pOutput*)

To get Host Dev Info unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.111 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	*pespLen*	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

9.3.2.112 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	*pespLen*	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

9.3.2.113 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	*pespLen*	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

9.3.2.114 int unpack_dms_SLQSSwiSetHostDevInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetHostDevInfo_t * *pOutput*)

To get Host Dev Info unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.115 int unpack_dms_SLQSSwiSetOSInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSSwiSetOSInfo_t * *pOutput*)

To get Host OS Info unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.116 int unpack_dms_SLQSUIMGetState (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_SLQSUIMGetState_t * *pOutput*)

To get UIM Get State unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.117 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	*pespLen*	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

9.3.2.118 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	*pespLen*	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

9.3.2.119 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	*pespLen*	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

9.3.2.120 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	*pespLen*	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

9.3.2.121 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	*pespLen*	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

9.3.2.122 int unpack_dms_UIMSetPINProtection (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMSetPIN-Protection_t * *pOutput*)

To get UIM Set PIN protection unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.123 int unpack_dms_UIMUnblockControlKey (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_UIMUnblock-ControlKey_t * *pOutput*)

To get UIM Unblock Control Key unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.3.2.124 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	*pespLen*	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

9.3.2.125 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	*pespLen*	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

9.3.2.126 int unpack_dms_ValidateSPC (uint8_t * *pResp*, uint16_t *respLen*, unpack_dms_ResetToFactoryDefaults_t * *pOutput*)

To get UIM Unblock Control Key unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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 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)

9.4.1 Macro Definition Documentation

9.4.1.1 #define FMS_FW_PRI_BUILD_MATCH_LEN 11

9.4.1.2 #define FMS_GOBI_LISTENTRIES_MAX 2

9.4.1.3 #define FMS_GOBI_MBN_BUILD_ID_STR_LEN 100

9.4.1.4 #define FMS_GOBI_MBN_IMG_ID_STR_LEN 16

9.4.1.5 #define FMS_IMAGE_ID_BUILD_ID_LEN 32

9.4.1.6 #define FMS_IMAGE_ID_IMG_ID_LEN 16

9.4.1.7 #define FMS_IMAGE_ID_MAX_ENTRIES 2

9.4.1.8 #define FMS_IMAGE_ID_PRI_IMGTYPE 0x01

9.4.1.9 #define FMS_MAX_IMAGE_ID_ELEMENT 50

9.4.1.10 #define FMS_MAX_IMAGE_PREFERENCE_IMAGE_SIZE 255

9.4.2 Function Documentation

9.4.2.1 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

9.4.2.2 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
in	*reqArg*	request parameter

9.4.2.3 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
in	*reqArg*	request parameter

9.4.2.4 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
in	*reqArg*	request parameter

9.4.2.5 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	*pespLen*	qmi response length
out	*pOutput*	unpacked response

9.4.2.6 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	*pespLen*	qmi response length
out	*pOutput*	unpacked response

9.4.2.7 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	*pespLen*	qmi response length
out	*pOutput*	unpacked response

9.5 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_9X30 =1 }

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)

9.5.1 Detailed Description

Filename: lite-fw.h

Purpose: Global definitions used inside the SDK

Copyright: © 2016 Sierra Wireless Inc., all rights reserved

9.5.2 Macro Definition Documentation

9.5.2.1 #define FIRMWARE_INFO_STRING_SIZE 20

9.5.2.2 #define IMG_MASK_CLEAR (0x000)

9.5.2.3 #define IMG_MASK_GENERIC (0xFFFF)

9.5.2.4 #define IMG_MASK_MDM (1<<0)

9.5.2.5 #define IMG_MASK_PRI (1<<1)

9.5.2.6 #define libSDP_BuildImagesPreferenceRequest litefw_BuildImagesPreferenceRequest

9.5.2.7 #define libSDP_CalculateImageMask litefw_CalculateImageMask

9.5.2.8 #define LIBSDP_CARRIER_PACKAGE_SKU LITEFW_CARRIER_PACKAGE_SKU

9.5.2.9 #define libSDP_CheckValidFirmwareInfo litefw_CheckValidFirmwareInfo

9.5.2.10 #define libSDP_DownloadFW litefw_DownloadFW

9.5.2.11 #define libSDP_ExtractFirmwareParametersByPath litefw_ExtractFirmwareParametersByPath

9.5.2.12 #define libSDP_getFileType litefw_getFileType

9.5.2.13 #define libSDP_GetModelFamily litefw_GetModelFamily

9.5.2.14 #define libSDP_GetVersion litefw_GetVersion

9.5.2.15 #define libsdp_set_log_func litefw_set_log_func

9.5.2.16 #define libsdp_SetReadBlockSize litefw_SetReadBlockSize

9.5.2.17 #define LIBSDP_SKU_STRING_LENGTH LITEFW_SKU_STRING_LENGTH

9.5.2.18 #define LITEFW_CARRIER_PACKAGE_SKU "9999999"

9.5.2.19 #define LITEFW_SKU_STRING_LENGTH 7

9.5.3 Typedef Documentation

9.5.3.1 typedef litefw_FirmwareInfo libSDP_FirmwareInfo

9.5.3.2 typedef litefwlogger libsdplogger

9.5.3.3 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.

9.5.3.4 typedef void(∗ litefwlogger)(uint8_t lvl, const char ∗buff)

This Custom Log prototype.

Parameters

lvl[IN]	• Log level.
buff[IN]	• Log String.

Returns

none

9.5.4 Enumeration Type Documentation

9.5.4.1 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

9.5.4.2 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

9.5.4.3 enum litefw_Models

Enumerator

eModel_Unknown

eModel_9X15

eModel_9X30

9.5.5 Function Documentation

9.5.5.1 int litefw_BuildImagesPreferenceRequest (litefw_FirmwareInfo *info,* pack_fms_SetImagesPreference_t * *pack*)

This API Build Image Preference Request Using Firmware Information.

Parameters

info[IN]	• See litefw_FirmwareInfo for more information.
pack[OUT]	• 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

9.5.5.2 int litefw_CalculateImageMask (unpack_fms_SetImagesPreference_t *SetPrefRspFromModem*)

This API Calculate Image Mask for Firmware Download.

Parameters

SetPrefRsp-FromModem[IN]	• See unpack_fms_SetImagesPreference_t for more information.

Returns

Image Mask

- IMG_MASK_MDM | IMG_MASK_PRI

9.5.5.3 int litefw_CheckValidFirmwareInfo (litefw_FirmwareInfo *info*)

This API Check Valid Firmware Information to build Image Prefernce Request.

Parameters

info[IN]	• See litefw_FirmwareInfo for more information.

Returns

0 on success, -1 error value otherwise

9.5.5.4 unsigned int litefw_DownloadFW (char * *pImagePath*, char * *szTTYPath*, int *iFWImageType*, int *image_mask*, int *iModelFamily*)

This API Download Firmware.

Parameters

pImagePath[IN]	• Firmware Folder Path.
szTTYPath[IN]	• QDL Device Path.
iFWImageType[IN]	• Firmware Type. • See litefw_Fw_Type
image_mask[IN]	• Image Mask. – IMG_MASK_MDM | IMG_MASK_PRI
iModelFamily[IN]	• 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

9.5.5.5 int litefw_ExtractFirmwareParametersByPath (char * *pImagePath,* litefw_FirmwareInfo * *info*)

This API Extrace Firmware Parameters From Path.

Parameters

pImagePath[IN]	• Firmware Folder Path.
litefw_Firmware-Info[OUT]	• 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

9.5.5.6 int litefw_getFileType (char * *szPath*)

This API Get File Type By Path

Parameters

szPath[IN]	• 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

9.5.5.7 int litefw_GetModelFamily (char * *pModelString*)

This API Get Model Famliy from a model string.

Parameters

pModelString[IN]	• Model String.

Returns

eModel_Unknown on Error, litefw_Models value otherwise

See Also

See litefw_Models for values

9.5.5.8 char* litefw_GetVersion ( )

This API Get Lib SDP Version.

Parameters

No;	

Returns

Version String

9.5.5.9 int litefw_set_log_func (litefwlogger *func*)

This API Set Custom Log function.

Parameters

func[IN]	• See litefwlogger for more information.

Returns

none

9.5.5.10 void litefw_SetReadBlockSize (unsigned long *lBlockSize*)

This API Set Read BlockSize.

Parameters

iBlockSize[IN]	• Firmware Read Block Size.

Returns

none

9.6 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

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

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)

9.6.1 Macro Definition Documentation

9.6.1.1 #define LOC_UINT8_MAX_STRING_SZ 255

9.6.1.2 #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.

9.6.1.3 #define LOCEVENTMASKENGINESTATE 0x00000080

The control point must enable this mask to receive engine state report event indications.

9.6.1.4 #define LOCEVENTMASKFIXSESSIONSTATE 0x00000100

The control point must enable this mask to receive fix session status report event indications.

9.6.1.5 #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.

9.6.1.6 #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.

9.6.1.7 #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.

9.6.1.8 #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).

9.6.1.9 #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.

9.6.1.10 #define LOCEVENTMASKINJECTPOSITIONREQ 0x00000040

The control point must enable this mask to receive position injection request event indications.

9.6.1.11 #define LOCEVENTMASKINJECTPREDICTEDORBITSREQ 0x00000020

The control point must enable this mask to receive predicted orbits request event indications.

9.6.1.12 #define LOCEVENTMASKINJECTTIMEREQ 0x00000010

The control point must enable this mask to receive time injection request event indications.

9.6.1.13 #define LOCEVENTMASKINJECTWIFIAPDATAREQ 0x00200000

The control point must enable this mask to receive Wi-Fi Access Point (AP) data inject request event indications.

9.6.1.14 #define LOCEVENTMASKINVALIDVALUE 0xFFFFFFFF

Invalid Event Mask

9.6.1.15 #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.

9.6.1.16 #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.

9.6.1.17 #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.

9.6.1.18 #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.

9.6.1.19 #define LOCEVENTMASKNINOTIFYVERIFYREQ 0x00000008

The control point must enable this mask to receive NI Notify/Verify request event indications.

9.6.1.20 #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.

9.6.1.21 #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.

9.6.1.22 #define LOCEVENTMASKPOSITIONREPORT 0x00000001

The control point must enable this mask to receive position report event indications.

9.6.1.23 #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.).

9.6.1.24 #define LOCEVENTMASKSETSPISTREAMINGREPORT 0x00001000

The control point must enable this mask to receive Stationary Position Indicator (SPI) streaming report indications.

9.6.1.25 #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.

9.6.1.26 #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.).

9.6.1.27 #define LOCEVENTMASKWIFIREQ 0x00000200

The control point must enable this mask to receive Wi-Fi position request event indications.

9.6.1.28 #define MAX_SENSOR_DATA_LEN 64

9.6.1.29 #define MAX_TEMP_DATA_LEN 64

9.6.2 Enumeration Type Documentation

9.6.2.1 anonymous enum

Enumerator

eQMI_LOC_SESS_STATUS_SUCCESS

eQMI_LOC_SESS_STATUS_IN_PROGRESS

eQMI_LOC_SESS_STATUS_FAILURE

eQMI_LOC_SESS_STATUS_TIMEOUT

9.6.3 Function Documentation

9.6.3.1 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	*pResp*	qmi response
in	*pespLen*	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

9.6.3.2 int pack_loc_EventRegister (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_EventRegister_t * *reqArg*)

Event Register pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.6.3.3 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	*pResp*	qmi response
in	*pespLen*	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

9.6.3.4 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	*pResp*	qmi response
in	*pespLen*	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

9.6.3.5 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	*pResp*	qmi response
in	*pespLen*	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

9.6.3.6 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

9.6.3.7 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

9.6.3.8 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

9.6.3.9 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

9.6.3.10 int pack_loc_Start (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_Start_t * *reqArg*)

LOC Start pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.6.3.11 int pack_loc_Stop (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_loc_Stop_t * *reqArg*)

Loc Stop pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.6.3.12 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	*pespLen*	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

9.6.3.13 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	*pespLen*	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

9.6.3.14 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	*pespLen*	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

9.6.3.15 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	*pespLen*	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

9.6.3.16 int unpack_loc_EventRegister (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_EventRegister_t * *pOutput*)

Event Register unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.6.3.17 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	*pespLen*	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

9.6.3.18 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	*pespLen*	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

9.6.3.19 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	*pespLen*	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

9.6.3.20 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	*pespLen*	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

9.6.3.21 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	*pespLen*	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

9.6.3.22 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	*pespLen*	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

9.6.3.23 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	*pespLen*	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

9.6.3.24 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

9.6.3.25 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

9.6.3.26 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

9.6.3.27 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

9.6.3.28 int unpack_loc_Start (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Start_t * *pOutput*)

Loc Start unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.6.3.29 int unpack_loc_Stop (uint8_t * *pResp*, uint16_t *respLen*, unpack_loc_Stop_t * *pOutput*)

Loc Stop unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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 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)

9.7.1 Macro Definition Documentation

9.7.1.1 #define NAS_IMSI_M_S1_LENGTH 7

9.7.1.2 #define NAS_IMSI_M_S2_LENGTH 3

9.7.1.3 #define NAS_MAX_DESCRIPTION_LENGTH 255

9.7.1.4 #define NAS_MAX_NUM_NETWORKS 30

9.7.1.5 #define NAS_MAX_SCC_RX_INFO_INSTANCES 255

9.7.1.6 #define NAS_NAM_NAME_LENGTH 12

9.7.1.7 #define NAS_OTA_MESSAGE_MAX_BUF_SIZE 2048

9.7.1.8 #define NAS_PLMN_LENGTH 3

9.7.1.9 #define NAS_SERVING_SYSTEM_INFO_MAX_RADIO_INTERFACE_LIST 255

9.7.2 Enumeration Type Documentation

9.7.2.1 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

9.7.2.2 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

9.7.2.3 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

9.7.2.4 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

9.7.3 Function Documentation

9.7.3.1 int pack_nas_GetACCOLC (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.2 int pack_nas_GetANAAAAuthenticationStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.3 int pack_nas_GetCDMANetworkParameters (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.4 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	*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

9.7.3.5 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	*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

9.7.3.6 int pack_nas_GetNetworkPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length

9.7.3.7 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	*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

9.7.3.8 int pack_nas_GetServingNetwork (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.9 int pack_nas_GetServingNetworkCapabilities (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.10 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

9.7.3.11 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	*pReq*	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

9.7.3.12 int pack_nas_PerformNetworkScan (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.13 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	*pReq*	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

9.7.3.14 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	*pReq*	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

9.7.3.15 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
in	*bEnable*	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

9.7.3.16 int pack_nas_SetNetworkPreference (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SetNetworkPreference_t * *reqArg*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length
in	*pack*	default prototype

9.7.3.17 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
in	*bEnable*	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

9.7.3.18 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	*pReq*	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

9.7.3.19 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	*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

9.7.3.20 int pack_nas_SlqsGetLTECphyCAInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	qmi request buffer
out	*pLen*	qmi request length

9.7.3.21 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

9.7.3.22 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	*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

9.7.3.23 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

9.7.3.24 int pack_nas_SLQSGetServingSystem (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.25 int pack_nas_SLQSGetSignalStrength (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, uint16_t *reqMask*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	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

9.7.3.26 int pack_nas_SLQSGetSysInfo (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.27 int pack_nas_SLQSGetSysSelectionPref (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.28 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	*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

9.7.3.29 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	*pReq*	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

9.7.3.30 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

9.7.3.31 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

9.7.3.32 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	*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

9.7.3.33 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	*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

9.7.3.34 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

9.7.3.35 int pack_nas_SLQSNasIndicationRegisterExt (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSNasIndicationRegisterExt_t * *pReqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	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

9.7.3.36 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	*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

9.7.3.37 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	*pReq*	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

9.7.3.38 int pack_nas_SLQSNasSwiModemStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.39 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	*pReq*	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

9.7.3.40 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	*pReq*	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

9.7.3.41 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	*pReq*	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

9.7.3.42 int pack_nas_SLQSSetSysSelectionPref (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_nas_SLQSSetSysSelectionPref_t * *pReqParam*)

Parameters

in,out	*pCtx*	qmi request context
out	*pReq*	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

9.7.3.43 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	*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

9.7.3.44 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	*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

9.7.3.45 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	*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

9.7.3.46 int pack_nas_SLQSSwiGetLteCQI (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.7.3.47 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

9.7.3.48 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	*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

9.7.3.49 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	*pReq*	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

9.7.3.50 int unpack_nas_GetACCOLC (uint8_t * *pResp*, uint16_t *respLen*, uint8_t * *pAccolc*)

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

9.7.3.51 int unpack_nas_GetANAAAAuthenticationStatus (uint8_t * *pResp*, uint16_t *respLen*, uint32_t * *pAuthStatus*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*authStatus*	auth status

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.52 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

9.7.3.53 int unpack_nas_GetHomeNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetHomeNetwork_t * *pOutput*)

get home network unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.54 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.55 int unpack_nas_GetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetNetworkPreference_t * *pOutput*)

9.7.3.56 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.57 int unpack_nas_GetServingNetwork (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetwork_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.58 int unpack_nas_GetServingNetworkCapabilities (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetServingNetworkCapabilities_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.59 int unpack_nas_GetSignalStrengths (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_GetSignalStrengths_t * *pOutput*)

get signal strengths unpack

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.60 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

9.7.3.61 int unpack_nas_PerformNetworkScan (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_PerformNetworkScan_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.62 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

9.7.3.63 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

9.7.3.64 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	*pOuput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.65 int unpack_nas_SetEventReportInd (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetEventReportInd_t * *pOutput*)

9.7.3.66 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

9.7.3.67 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	*pOuput*	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

9.7.3.68 int unpack_nas_SetNetworkPreference (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetNetwork-Preference_t * *pOutput*)

9.7.3.69 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

9.7.3.70 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	*pOuput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.71 int unpack_nas_SetServingSystemCallback_ind (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SetServingSystemCallback_ind_t * *pOutput*)

9.7.3.72 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

9.7.3.73 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.74 int unpack_nas_SlqsGetLTECphyCAInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SlqsGetLTECphyCAInfo_t * *pOutput*)

9.7.3.75 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

9.7.3.76 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.77 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

9.7.3.78 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.79 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.80 int unpack_nas_SLQSGetSysInfo (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysInfo_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.81 int unpack_nas_SLQSGetSysSelectionPref (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSGetSysSelectionPref_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.82 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

9.7.3.83 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

9.7.3.84 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

9.7.3.85 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

9.7.3.86 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.87 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.88 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

9.7.3.89 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

9.7.3.90 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	*pOuput*	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

9.7.3.91 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	*pOuput*	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

9.7.3.92 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.93 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

9.7.3.94 int unpack_nas_SLQSNasSwiModemStatus (uint8_t * *pResp*, uint16_t *respLen*, unpack_nas_SLQSNasSwiModemStatus_t * *pOutput*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length
out	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.95 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	*pOuput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.96 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

9.7.3.97 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.98 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	*pOuput*	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

9.7.3.99 int unpack_nas_SLQSSetBandPreference (uint8_t * *pResp*, uint16_t *respLen*)

Parameters

in	*pResp*	qmi response from modem
in	*respLen*	qmi response length

9.7.3.100 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

9.7.3.101 int unpack_nas_SLQSSetSysSelectionPref (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

9.7.3.102 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	*pOuput*	indication unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.103 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.104 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.105 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.106 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.107 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.108 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.7.3.109 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

9.8 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

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)

9.8.1 Function Documentation

9.8.1.1 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	*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

9.8.1.2 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	*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

9.8.1.3 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	*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

9.8.1.4 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	*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

9.8.1.5 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	*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

9.8.1.6 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	*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

9.8.1.7 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.1.8 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.1.9 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.8.1.10 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

9.8.1.11 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

9.8.1.12 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 qaGobiApiTableBandClasses.h File Reference

Network Access Service API Band Classes table.

9.9.1 Detailed Description

Network Access Service API Band Classes table.

9.9.2 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

9.9.2.1 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

9.10 qaGobiApiTableCallControlReturnReasons.h File Reference

Call Control Return Reasons table.

9.10.1 Detailed Description

Call Control Return Reasons table.

9.10.2 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

9.11 qaGobiApiTableCallEndReasons.h File Reference

Wireless Data Service Call End Reasons.

9.11.1 Detailed Description

Wireless Data Service Call End Reasons.

9.11.2 Call end reason codes (Code - Reason)

9.11.2.1 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

9.11.2.2 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

9.11.2.3 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

9.11.2.4 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

9.11.2.5 call end reason type

- 1 - Mobile IP
- 2 - Internal
- 3 - Call Manager defined
- 6 - 3GPP specification defined
- 7 - PPP
- 8 - EHRPD
- 9 - IPv6

9.11.2.6 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

9.11.2.7 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

9.11.2.8 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
- 525 - 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

9.11.2.9 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

9.11.2.10 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

9.11.2.11 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

9.11.2.12 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

9.12 qaGobiApiTableCarrierCodes.h File Reference

Carrier Codes table.

9.12.1 Detailed Description

Carrier Codes table.

9.12.2 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

9.13 qaGobiApiTableCodingScheme.h File Reference

Data Coding Scheme.

Macros

- #define __GOBI_API_CODING_SCHEME_H__

9.13.1 Detailed Description

Data Coding Scheme.

9.13.2 Call Control Result Reasons (Value - Name - Description)

9.13.2.1 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

9.13.3 Coding Group Bits 7..4(0001)

9.13.3.1 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.

9.13.4 Coding Group Bits 7..4(0010)

9.13.4.1 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

9.13.5 Coding Group Bits 7..4(0011)

9.13.5.1 use of bits 3..0

- 0000..1111 Reserved for other languages using the GSM 7 bit default alphabet, with unspecified handling at the MS

9.13.6 Coding Group Bits 7..4(01xx)

9.13.6.1 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

9.13.7 Coding Group Bits 7..4(1001)

9.13.7.1 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

9.13.8 Coding Group Bits 7..4(1010..1101)

9.13.8.1 Reserved coding groups

9.13.9 Coding Group Bits 7..4(1110)

9.13.9.1 Defined by the WAP Forum

9.13.10 Coding Group Bits 7..4 (1111)

9.13.10.1 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

9.13.11 Macro Definition Documentation

9.13.11.1 #define __GOBI_API_CODING_SCHEME_H__

9.14 qaGobiApiTableGpsCapabilityCodes.h File Reference

Position Determination Service API GPS Capability Codes.

9.14.1 Detailed Description

Position Determination Service API GPS Capability Codes.

9.14.2 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

9.15 qaGobiApiTablePowerModes.h File Reference

Device Management Service API Power Modes table.

9.15.1 Detailed Description

Device Management Service API Power Modes table.

9.15.2 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

9.16 qaGobiApiTableRadioInterfaces.h File Reference

Network Access Service API Radio Interfaces table.

9.16.1 Detailed Description

Network Access Service API Radio Interfaces table.

9.16.2 Radio interface

9.16.2.1 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

9.17 qaGobiApiTableRegionCodes.h File Reference

Region Codes table.

9.17.1 Detailed Description

Region Codes table.

9.17.2 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

9.18 qaGobiApiTableServiceOptions.h File Reference

Voice Service Options.

9.18.1 Detailed Description

Voice Service Options.

9.18.2 Service Option codes (Code - Reason)

9.18.2.1 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

9.19 qaGobiApiTableSupServiceInfoClasses.h File Reference

Voice Supplementary Service Information Classes.

9.19.1 Detailed Description

Voice Supplementary Service Information Classes.

9.19.2 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

9.20 qaGobiApiTableSwiAudio.h File Reference

Swi Audio related tables.

9.20.1 Detailed Description

Swi Audio related tables.

9.20.2 ACDB Device (Device ID - description)

- 0 - Vehicle HF
- 1 - Handset
- 2 - TTY
- 3 - USB
- 4 - NA

9.20.3 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

9.21 qaGobiApiTableSwiOMADMUpdateCompleteStatus.h File Reference

Update Complete Status table.

9.21.1 Detailed Description

Update Complete Status table.

9.21.2 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

9.22 qaGobiApiTableVoiceCallEndReasons.h File Reference

Voice Service Call and supplementary services end reasons.

9.22.1 Detailed Description

Voice Service Call and supplementary services end reasons.

9.22.2 Voice Call and supplementary services end reason codes (Code - Reason)

9.22.2.1 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

9.22.2.2 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

9.22.2.3 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

9.22.2.4 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

9.22.2.5 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

9.22.2.6 reject causes

- 226 - Timer T303 is expired
- 227 - CNM MM release is pending

9.22.2.7 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

9.22.2.8 reject causes

- 237 - Service option is not supported

9.22.2.9 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

9.23 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 }

9.23.1 Enumeration Type Documentation

9.23.1.1 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

9.23.1.2 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

9.24 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)

9.24.1 Macro Definition Documentation

9.24.1.1 #define LITEQMI_MAX_QOS_FILTERS 25

9.24.1.2 #define LITEQMI_MAX_QOS_FLOW_PER_APN_STATS 10

9.24.1.3 #define LITEQMI_MAX_QOS_FLOWS 8

9.24.2 Function Documentation

9.24.2.1 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• 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

9.24.2.2 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes

pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.24.2.3 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.24.2.4 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.24.2.5 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.24.2.6 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.24.2.7 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.24.2.8 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

pResp[IN]	• Response from modem
respLen[IN]	• 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

9.24.2.9 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.24.2.10 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.24.2.11 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.24.2.12 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.25 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)

9.25.1 Function Documentation

9.25.1.1 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	*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

9.25.1.2 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.25.1.3 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

9.25.1.4 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

9.26 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)

9.26.1 Function Documentation

9.26.1.1 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	*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

9.26.1.2 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.26.1.3 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

9.26.1.4 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

9.27 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)

9.27.1 Macro Definition Documentation

9.27.1.1 #define MAX_CDMA_ENC_MO_TXT_MSG_SIZE 255

9.27.1.2 #define MAX_MS_TRANSFER_ROUTE_MSG 256

9.27.1.3 #define MAX_MSC_ADDRESS_SIZE 256

9.27.1.4 #define MAX_MSE_TWS_MSG 1254

9.27.1.5 #define MAX_SMS_LIST_SIZE 255

9.27.1.6 #define MAX_SMS_MESSAGE_SIZE 2048

9.27.1.7 #define SMS_CONFIG_LEN 0x05

9.27.1.8 #define SMS_MAX_SMS_ROUTES 0x0A

9.27.1.9 #define SMS_NUM_OF_SET 0xFF

9.27.1.10 #define SMSC_TYPE_LEN 0x03

9.27.2 Typedef Documentation

9.27.2.1 typedef struct sMSCAddress sMSCAddressInfo

Parameters

length	• Number of sets of following element
data	• SMSC address

9.27.2.2 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

9.27.2.3 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

9.27.2.4 typedef struct sMSMessageMode sMSMessageModeInfo

Parameters

messageMode	Message Mode

9.27.2.5 typedef struct sMSMTMessage sMSMTMessageInfo

Parameters

storageType	memory storage 0x00-UIM 0x01-NV
messageIndex	MT Message index

9.27.2.6 typedef struct sMSOnIMS sMSOnIMSInfo

Parameters

smsOnIMS	SMS on IMS

9.27.2.7 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

9.27.3 Enumeration Type Documentation

9.27.3.1 enum eqmiCbkSetStatus

Enumerator

LITEQMI_QMI_CBK_PARAM_RESET

LITEQMI_QMI_CBK_PARAM_SET

LITEQMI_QMI_CBK_PARAM_NOCHANGE

9.27.4 Function Documentation

9.27.4.1 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

9.27.4.2 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

9.27.4.3 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

9.27.4.4 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

9.27.4.5 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

9.27.4.6 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

9.27.4.7 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

9.27.4.8 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

9.27.4.9 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

9.27.4.10 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

9.27.4.11 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

9.27.4.12 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

9.27.4.13 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

9.27.4.14 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

9.27.4.15 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

9.27.4.16 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

9.27.4.17 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

9.27.4.18 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

9.27.4.19 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

9.27.4.20 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

9.27.4.21 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

9.27.4.22 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

9.27.4.23 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

9.27.4.24 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	*pespLen*	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

9.27.4.25 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	*pespLen*	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

9.27.4.26 int unpack_sms_SendSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SendSMS_t * *pOutput*)

send sms unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.27.4.27 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	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.28 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	*pespLen*	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

9.27.4.29 int unpack_sms_SetSMSCAddress (uint8_t * *pResp*, uint16_t *respLen*)

Set the SMS center address unpack.

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.30 int unpack_sms_SLQSDeleteSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSDeleteSMS_t * *pOutput*)

delete sms unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.31 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	*pespLen*	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

9.27.4.32 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	*pespLen*	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

9.27.4.33 int unpack_sms_SLQSGetSMS (uint8_t * *pResp*, uint16_t *respLen*, unpack_sms_SLQSGetSMS_t * *pOutput*)

get sms unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.27.4.34 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	*pespLen*	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

9.27.4.35 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	*pespLen*	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

9.27.4.36 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	*pespLen*	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

9.27.4.37 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	*pespLen*	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

9.27.4.38 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	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.39 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	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.40 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	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.41 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	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.42 int unpack_sms_SLQSSetSmsBroadcastConfig (uint8_t * *pResp*, uint16_t *respLen*)

Sets the information about the SMS BroadcastConfiguration unpack.

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.43 int unpack_sms_SLQSSetSmsStorage (uint8_t * *pResp*, uint16_t *respLen*)

Sets the SMS Storage on the device unpack.

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.44 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	*pespLen*	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

9.27.4.45 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	*pespLen*	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

9.27.4.46 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	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.27.4.47 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	*pespLen*	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

9.27.4.48 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	*pespLen*	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

9.28 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)

9.28.1 Macro Definition Documentation

9.28.1.1 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.28.1.2 #define MAX_AVMS_SETTINGS_RETRY_TIMER_NUMBER 8

9.28.1.3 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_DESC_LENGTH 1024

9.28.1.4 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_NAME_LENGTH 128

9.28.1.5 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_BINARYUPDATESESSIONINFO_VERSION_LENGTH 128

9.28.1.6 #define MAX_PACK_SWI_AVMS_SESSIONGETINFO_CONFIG_ALERT_MSG_LENGTH 200

9.28.1.7 #define MAX_PACK_SWI_AVMS_SET_SETTING_APN_LENGTH 49

9.28.1.8 #define MAX_PACK_SWI_AVMS_SET_SETTING_CONNECTION_RETRY_TIMMERS 8

9.28.1.9 #define MAX_PACK_SWI_AVMS_SET_SETTING_PWD_LENGTH 29

9.28.1.10 #define MAX_PACK_SWI_AVMS_SET_SETTING_UNAME_LENGTH 29

9.28.2 Typedef Documentation

9.28.2.1 typedef struct _PackSwiAvmsSetSettingsAPNInfo PackSwiAvmsSetSettingsAPNInfo

Parameters

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

9.28.2.2 typedef struct _PackSwiAvmsSetSettingsConnectionRetryTimers PackSwiAvmsSetSettings-ConnectionRetryTimers

9.28.2.3 typedef struct _PackSwiAvmsSetSettingsPeriodInfo PackSwiAvmsSetSettingsPeriodInfo

Parameters

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

9.28.2.4 typedef struct _PackSwiAVMSSettingsAPNInfo PackSwiAVMSSettingsAPNInfo

9.28.2.5 typedef struct _PackSwiAVMSSettingsConnectionRetryTimers PackSwiAVMSSettingsConnection-RetryTimers

9.28.2.6 typedef struct _PackSwiAVMSSettingsPeriodsInfo PackSwiAVMSSettingsPeriodsInfo

9.28.2.7 typedef struct _unpack_swiavms_SLQSAvMSEventReportInd_t unpack_swiavms_SLQSAvMSEvent-ReportInd_t

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

Parameters

pBinaryUpdate-SessionInfo[OU-T]	• See UnpackSwiAvmsEventReportBinaryUpdateSessionInfo for more information
pNotification[O-UT]	• See UnpackSwiAvmsEventReportConfig for more information
pPackageID[OU-T]	• See UnpackSwiAvmsEventReportPackageID for more information
pConnection-Request[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
UnpackSwiAvmsEventReportSessionType[OUT]	• See pSessionType for more information
pWAMSParaChanged[OUT]	• See UnpackSwiAvmsEventReportHTTPStatus for more information
pHTTPStatus[OUT]	• See pSessionType for more information
resultcode	- Tlv Result Code.

9.28.2.8 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.

9.28.2.9 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

9.28.2.10 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

9.28.2.11 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

9.28.2.12 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

9.28.2.13 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

9.28.2.14 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

9.28.2.15 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

9.28.2.16 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

9.28.2.17 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

9.28.2.18 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

9.28.3 Function Documentation

9.28.3.1 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• 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

9.28.3.2 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.28.3.3 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes

pLen[IN/OUT]	• 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

9.28.3.4 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• 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

9.28.3.5 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.28.3.6 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.28.3.7 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.28.3.8 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.28.3.9 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pResponse[OUT]	• 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

9.28.3.10 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pResponse[OUT]	• 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

9.28.3.11 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pResponse[OUT]	• 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

9.28.3.12 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pResponse[OUT]	• 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

9.28.3.13 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pResponse[OUT]	• 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

9.28.3.14 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

pResp[IN]	• Response from modem

respLen[IN]	• Length of pResp from modem
pResponse[OUT]	• 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

9.28.3.15 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pResponse[OUT]	• 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

9.28.3.16 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem

pResponse[OUT]	• 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

9.29 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

9.29.1 Detailed Description

SWI data types.

9.29.2 Macro Definition Documentation

9.29.2.1 #define QMI_NO_LTE_FW_SUPPORT 0

9.29.2.2 #define QMI_TLV_PLACEHOLDER 0x8F

9.29.2.3 #define SWI_API

9.29.2.4 #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.

9.29.3 Typedef Documentation

9.29.3.1 typedef int BOOL

9.29.3.2 typedef unsigned char BYTE

9.29.3.3 typedef char CHAR

9.29.3.4 typedef float FLOAT

9.29.3.5 typedef signed int INT32

9.29.3.6 typedef signed char INT8

9.29.3.7 typedef const char* LPCSTR

9.29.3.8 typedef signed short SHORT

9.29.3.9 typedef unsigned int ULONG

9.29.3.10 typedef unsigned long long ULONGLONG

9.29.3.11 typedef unsigned short USHORT

9.29.3.12 typedef unsigned short WORD

9.30 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)

9.30.1 Function Documentation

9.30.1.1 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

9.30.1.2 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.1.3 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	*pOuput*	response unpacked

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.30.1.4 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

9.31 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)

9.31.1 Macro Definition Documentation

9.31.1.1 #define LITEQMI_MAX_SWIOMA_STR_LEN 255

9.31.2 Function Documentation

9.31.2.1 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• 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

9.31.2.2 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

pCtx[OUT]	• See pack_qmi_t for more information

pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.31.2.3 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.31.2.4 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• 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

9.31.2.5 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes

pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.31.2.6 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.31.2.7 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

pCtx[OUT]	• See pack_qmi_t for more information
pReqBuf[IN/OUT]	• Buffer for packed QMI command to be provided by the host application • Minimum expected size is 2048 bytes
pLen[IN/OUT]	• On input, size of pReqBuf • On ouptut, number of bytes actually packed
reqParam[IN]	• 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

9.31.2.8 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

pResp[IN]	• Response from modem
respLen[IN]	• 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

9.31.2.9 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.31.2.10 int unpack_swioma_SLQSOMADMCancelSession (uint8_t * *pResp*, uint16_t *respLen*)

Function to pack cancel OMA-DM session command This maps to SLQSOMADMCancelSession

Parameters

pResp[IN]	• Response from modem
respLen[IN]	• 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

9.31.2.11 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.31.2.12 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.31.2.13 int unpack_swioma_SLQSOMADMSendSelection (uint8_t * *pResp*, uint16_t *respLen*)

Function to unpack OMA-DM send selection command This maps to SLQSOMADMSendSelection2

Parameters

pResp[IN]	• Response from modem
respLen[IN]	• 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

9.31.2.14 int unpack_swioma_SLQSOMADMSetSettings (uint8_t * *pResp*, uint16_t *respLen*)

Function to unpack OMA-DM set settings command This maps to SLQSOMADMSetSettings3

Parameters

pResp[IN]	• Response from modem
respLen[IN]	• 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

9.31.2.15 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

pResp[IN]	• Response from modem
respLen[IN]	• Length of pResp from modem
pOutput[OUT]	• 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

9.32 SWIWWANCMAPI.h File Reference

9.33 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)

9.33.1 Macro Definition Documentation

9.33.1.1 #define MAX_DESCRIPTION_LENGTH 255

9.33.1.2 #define MAX_ICCID_LENGTH 255

9.33.1.3 #define MAX_NO_OF_APPLICATIONS 10

9.33.1.4 #define MAX_NO_OF_SLOTS 5

9.33.1.5 #define MAX_SLOTS_STATUS 255

9.33.1.6 #define UIM_MAX_ACTIVE_PERS_FEATURES 12

9.33.1.7 #define UIM_MAX_CONTENT_LENGTH 1024

9.33.1.8 #define UIM_MAX_DESCRIPTION_LENGTH 255

9.33.1.9 #define UIM_MAX_NO_OF_APPLICATIONS 10

9.33.1.10 #define UIM_MAX_NO_OF_SLOTS 5

9.33.1.11 #define UIM_UINT8_MAX_STRING_SZ 255

9.33.2 Function Documentation

9.33.2.1 int pack_uim_ChangePin (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* pack_uim_ChangePin_t * *reqArg*)

Change Pin pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.33.2.2 int pack_uim_GetCardStatus (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* void * *reqArg*)

Get Card Status pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.33.2.3 int pack_uim_ReadTransparent (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*, pack_uim_ReadTransparent_t * *reqArg*)

SLQS ReadTransparent pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.33.2.4 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.33.2.5 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	*pReq*	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

9.33.2.6 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	*pReq*	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

9.33.2.7 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.33.2.8 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	*pReq*	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

9.33.2.9 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	*pReq*	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

9.33.2.10 int pack_uim_SLQSUIMGetSlotsStatus (pack_qmi_t * *pCtx*, uint8_t * *pReqBuf*, uint16_t * *pLen*)

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

9.33.2.11 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	*pReq*	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

9.33.2.12 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	*pReq*	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

9.33.2.13 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	*pReq*	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

9.33.2.14 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	*pReq*	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

9.33.2.15 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	*pReq*	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

9.33.2.16 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	*pReq*	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

9.33.2.17 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	*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

9.33.2.18 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	*pReq*	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

9.33.2.19 int pack_uim_UnblockPin (pack_qmi_t * *pCtx,* uint8_t * *pReqBuf,* uint16_t * *pLen,* pack_uim_UnblockPin_t * *reqArg*)

Unblock Pin pack

Parameters

in	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.33.2.20 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	*pResp*	qmi response
in	*pespLen*	qmi response length
out	*pOutput*	unpacked response
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

9.33.2.21 int unpack_uim_ChangePin (uint8_t * *pResp,* uint16_t *respLen,* unpack_uim_ChangePin_t * *pOutput*)

Change Pin unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.33.2.22 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	*pespLen*	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

9.33.2.23 int unpack_uim_ReadTransparent (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_ReadTransparent_t * *pOutput*)

SLQS ReadTransparent unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.33.2.24 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	*pespLen*	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

9.33.2.25 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	*pespLen*	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

9.33.2.26 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	*pespLen*	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

9.33.2.27 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	*pespLen*	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

9.33.2.28 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	*pespLen*	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

9.33.2.29 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	*pespLen*	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

9.33.2.30 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	*pespLen*	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

9.33.2.31 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

9.33.2.32 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

9.33.2.33 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

9.33.2.34 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

9.33.2.35 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	*pespLen*	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

9.33.2.36 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

9.33.2.37 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

9.33.2.38 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	*pespLen*	qmi response length

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.33.2.39 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	*pespLen*	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

9.33.2.40 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

9.33.2.41 int unpack_uim_UnblockPin (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_UnblockPin_t * *pOutput*)

Unblock Pin unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.33.2.42 int unpack_uim_VerifyPin (uint8_t * *pResp*, uint16_t *respLen*, unpack_uim_VerifyPin_t * *pOutput*)

Verify Pin unpack

Parameters

in	*pResp*	qmi response
in	*pespLen*	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

9.34 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

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)

9.34.1 Macro Definition Documentation

9.34.1.1 #define BARRING_PASSWORD_LENGTH 4

9.34.1.2 #define MAX_VOICE_CALL_NO_LEN 81

9.34.1.3 #define MAX_VOICE_DESCRIPTION_LENGTH 255

9.34.1.4 #define MAXVOICEUSSDLENGTH 182

9.34.1.5 #define VOICE_MAX_NO_OF_CALLS 20

9.34.2 Function Documentation

9.34.2.1 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.2 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

9.34.2.3 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.4 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.5 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.6 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.7 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	*pReq*	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

9.34.2.8 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.9 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	*pReq*	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

9.34.2.10 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	*pReq*	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

9.34.2.11 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	*pReq*	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

9.34.2.12 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	*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

9.34.2.13 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	*pReq*	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

9.34.2.14 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	*pReq*	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

9.34.2.15 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	*pReq*	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

9.34.2.16 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	*pReq*	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

9.34.2.17 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	*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

9.34.2.18 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	*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

9.34.2.19 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	*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

9.34.2.20 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	*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

9.34.2.21 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	*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

9.34.2.22 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.23 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.24 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	*pReq*	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

9.34.2.25 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.26 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	*pReq*	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

9.34.2.27 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	*pReq*	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

9.34.2.28 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	*pReq*	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

9.34.2.29 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.34.2.30 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	*pReq*	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

9.34.2.31 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	*pReq*	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

9.34.2.32 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	*pReq*	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

9.34.2.33 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

9.34.2.34 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

9.34.2.35 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

9.34.2.36 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

9.34.2.37 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

9.34.2.38 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

9.34.2.39 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

9.34.2.40 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

9.34.2.41 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

9.34.2.42 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

9.34.2.43 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

9.34.2.44 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

9.34.2.45 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

9.34.2.46 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

9.34.2.47 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

9.34.2.48 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

9.34.2.49 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

9.34.2.50 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

9.34.2.51 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

9.34.2.52 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

9.34.2.53 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

9.34.2.54 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

9.34.2.55 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

9.34.2.56 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

9.34.2.57 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

9.34.2.58 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

9.34.2.59 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

9.34.2.60 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

9.34.2.61 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

9.34.2.62 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

9.34.2.63 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

9.34.2.64 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

9.35 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_SLQSSetDHCPv4ClientConfig_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

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)

9.35.1 Macro Definition Documentation

9.35.1.1 #define BYT_STAT_STAT_MASK 0X000000C0

9.35.1.2 #define IPV6_ADDRESS_ARRAY_SIZE 8

9.35.1.3 #define MAX_WDS_3GPP_CONF_LTE_ATTACH_PROFILE_LIST_SIZE 24

9.35.1.4 #define PACK_WDS_IPV4 4

9.35.1.5 #define PACK_WDS_IPV6 6

9.35.1.6 #define WDS_DHCP_MAX_NUM_OPTIONS 30

9.35.1.7 #define WDS_DHCP_OPTION_DATA_BUF_SIZE 2048 /* current max size of raw message in SDK process is 2048 */

9.35.2 Typedef Documentation

9.35.2.1 typedef union unpackWdsProfileParam UnpackQmiProfileInfo

9.35.3 Function Documentation

9.35.3.1 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	*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

9.35.3.2 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

9.35.3.3 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	*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

Note

PDN Specific: Yes

9.35.3.4 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	*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

Note

PDN Specific: No

9.35.3.5 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	*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

Note

PDN Specific: Yes

9.35.3.6 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	*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

Note

PDN Specific: Yes

9.35.3.7 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	*pReq*	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

9.35.3.8 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	*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

Note

PDN Specific: No

9.35.3.9 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	*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

Note

PDN Specific: No

9.35.3.10 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	*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

Note

PDN Specific: No

9.35.3.11 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	*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

Note

PDN Specific: Yes

9.35.3.12 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	*pReq*	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

9.35.3.13 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	*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

Note

PDN Specific: Yes

9.35.3.14 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	*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

Note

PDN Specific: No

9.35.3.15 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	*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

Note

PDN Specific: Yes

9.35.3.16 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
out	*reqParam*	request parameter

Note

PDN Specific: No

9.35.3.17 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	*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

9.35.3.18 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	*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

Note

PDN Specific: Yes

9.35.3.19 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	*pReq*	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

9.35.3.20 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	*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

9.35.3.21 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	*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

9.35.3.22 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
out	*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

9.35.3.23 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	*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

9.35.3.24 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	*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

Note

PDN Specific: Yes

9.35.3.25 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	*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

Note

PDN Specific: Yes

9.35.3.26 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	*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

Note

PDN Specific: Yes

9.35.3.27 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	*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

Note

PDN Specific: Yes

9.35.3.28 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
out	*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

9.35.3.29 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
out	*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

9.35.3.30 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	*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

9.35.3.31 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

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

9.35.3.32 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	*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

9.35.3.33 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

9.35.3.34 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	*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

Note

PDN Specific: Yes

9.35.3.35 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
out	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.3.36 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

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

9.35.3.37 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	*pReq*	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

9.35.3.38 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	*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

Note

PDN Specific: Yes

9.35.3.39 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	*reqParam*	request parameter

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

see qmerrno.h for eQCWWAN_xxx error values

9.35.3.40 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	*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

9.35.3.41 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	*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

9.35.3.42 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	*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

9.35.3.43 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

9.35.3.44 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

9.35.3.45 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	*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

9.35.3.46 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	*pReq*	qmi request buffer
out	*pLen*	qmi request length
out	*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

9.35.3.47 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

9.35.3.48 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

9.35.3.49 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

9.35.3.50 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

9.35.3.51 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

9.35.3.52 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

9.35.3.53 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

9.35.3.54 int unpack_wds_GetDefaultProfileNum (uint8_t * *pResp*, uint16_t *respLen*, unpack_wds_GetDefaultProfile-Num_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

9.35.3.55 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

9.35.3.56 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

9.35.3.57 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

9.35.3.58 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

9.35.3.59 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

9.35.3.60 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

9.35.3.61 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

9.35.3.62 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

9.35.3.63 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.3.64 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

9.35.3.65 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

9.35.3.66 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
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

9.35.3.67 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

9.35.3.68 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

9.35.3.69 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

9.35.3.70 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.3.71 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
in	*pProfileId*	profile id pointer passed in req

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.3.72 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

9.35.3.73 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

9.35.3.74 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

9.35.3.75 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

9.35.3.76 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

9.35.3.77 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

9.35.3.78 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

9.35.3.79 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

9.35.3.80 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

Returns

eQCWWAN_ERR_NONE on success, eQCWWAN_xxx error value otherwise

See Also

See qmerrno.h for eQCWWAN_xxx error values

9.35.3.81 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

9.35.3.82 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

9.35.3.83 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

9.35.3.84 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

9.35.3.85 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

9.35.3.86 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

9.35.3.87 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

9.35.3.88 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

9.35.3.89 int unpack_wds_SLQSSSetDHCPv4ClientConfig (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

9.35.3.90 int unpack_wds_SLQSSSetLoopback (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

9.35.3.91 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

9.35.3.92 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

9.35.3.93 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

9.35.3.94 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

9.35.3.95 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

9.35.3.96 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, 585
_PackSwiAVMSSettingsAPNInfo, 29
  bAPNLength, 29
  bPWDLength, 29
  bUnameLength, 29
  pAPN, 29
  pPWD, 29
  pUname, 29
_PackSwiAVMSSettingsConnectionRetryTimers, 29
  Timers, 30
_PackSwiAVMSSettingsPeriodsInfo, 30
  max, 30
  min, 30
_PackSwiAvmsSetSettingsAPNInfo, 28
  bAPNLength, 28
  bPWDLength, 28
  bUnameLength, 28
  szAPN, 28
  szPWD, 28
  szUname, 28
_PackSwiAvmsSetSettingsConnectionRetryTimers, 28
  Timers, 29
_PackSwiAvmsSetSettingsPeriodInfo, 29
  ulMax, 29
  ulMin, 29
_UnpackSwiAvmsEventReportBinaryUpdateSession-
    Info, 32
  bBinaryType, 33
  bSerity, 33
  bState, 33
  bUserInputRequest, 33
  szDescription, 33
  szName, 33
  szVersion, 34
  TlvPresent, 34
  ulPkgDownloadComplete, 34
  ulPkgDownloadSize, 34
  wDescriptionLength, 34
  wNameLength, 34
  wUpdateCompeteStatus, 34
  wUserInputTimeout, 34
  wVersionLength, 34
_UnpackSwiAvmsEventReportConfig, 34
  bState, 34
  bUserInputRequest, 35
  szAlertMsg, 35
  TlvPresent, 35
  wAlertMsgLength, 35
  wUserInputTimeout, 35
_UnpackSwiAvmsEventReportConnectionRequest, 35
  bUserInputRequest, 35
  TlvPresent, 35
  wUserInputTimeout, 35
_UnpackSwiAvmsEventReportDataSessionStatus, 35
  bType, 36
  TlvPresent, 36
  wErrorCode, 36
_UnpackSwiAvmsEventReportHTTPStatus, 36
  TlvPresent, 37
  wHTTPStatus, 37
_UnpackSwiAvmsEventReportNotification, 38
  bNotification, 38
  TlvPresent, 38
  wSessionStatus, 39
_UnpackSwiAvmsEventReportPackageID, 39
  bPackageID, 39
  TlvPresent, 39
_UnpackSwiAvmsEventReportRegStatus, 39
  bRegStatus, 39
  TlvPresent, 39
_UnpackSwiAvmsEventReportSessionType, 39
  bType, 40
  TlvPresent, 40
_UnpackSwiAvmsEventReportWAMSParamChange, 40
  TlvPresent, 40
  wWamsChangeMask, 40
_litefw_FirmwareInfo_, 27
  szCarrier_str, 27
  szCarrierPriversion_str, 28
  szFwversion_str, 28
  szModelid_str, 28
  szPackageid_str, 28
  szSku_str, 28
_unpack_swiavms_SLQSAvMSEventReportInd_t, 30
  pConfig, 31
  pConnectionRequest, 31
  pNotification, 31
  pRegStatus, 31
  pSessionType, 31
  resultcode, 31
_unpack_swiavms_SLQSAvMSSessionGetInfo_t, 31
  pConfig, 32
  pNotification, 32
  resultcode, 32

AAASPI
  unpack_wds_GetMobileIPProfile_t, 579
AAAState

unpack_wds_GetMobileIPProfile_t, 579
ACT_CODE_MAX_SIZE
dms.h, 664
ALS
voice_getAllCallInformation, 625
AMSSString
unpack_dms_GetFirmwareRevision_t, 406
unpack_dms_GetFirmwareRevisions_t, 407
APNName
unpack_wds_SLQSGetRuntimeSettings_t, 590
accelTemp
pack_loc_SLQSLOCInjectSensorData_t, 248
acceleroData
pack_loc_SLQSLOCInjectSensorData_t, 248
acceleroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 248
AccessMac
nas_protocolSubtypeElement, 176
accolc
pack_nas_SetACCOLC_t, 252
accuracy
pack_pds_SetPDSDefaults_t, 276
ackIndicator
sMSTransferRouteMTMessage, 373
acqOrdeLen
nas_acqOrderPref, 101
acroamsetting
pack_wds_SetAutoconnect_t, 331
acsetting
pack_wds_SetAutoconnect_t, 331
actCode
pack_dms_ActivateAutomatic_t, 224
ActSetCnt
nas_NetworkStat1x, 162
ActSetPilotPN
nas_ActPilotPNElement, 101
ActSetPilotPNStrength
nas_ActPilotPNElement, 101
action
pack_nas_InitiateDomainAttach_t, 252
activated_ind
sms_qaQmi3GPP2BroadcastCfgInfo, 362
sms_qaQmi3GPPBroadcastCfgInfo, 363
activationState
pack_dms_SetActivationStatusCallback_t, 225
activationStatus
dms_ActivationStatusTlv, 49
ActivationStatusTlv
unpack_dms_SetEventReport_ind_t, 415
activeBandClass
nas_RFInfoTlv, 180
RFBandInfoElements, 350
activeChannel
nas_RFInfoTlv, 180
RFBandInfoElements, 350
activeInd
sms_messageWaitingInfoContent, 361
ActiveStatus
voice_CLIPResp, 617
voice_CLIRResp, 617
voice_CNAPResp, 618
voice_COLPResp, 619
voice_COLRResp, 619
ActiveTechPref
unpack_nas_GetNetworkPreference_t, 458
addr
unpack_qos_IPv4Addr_t, 500
unpack_qos_IPv6Addr_t, 500
address
unpack_wds_GetMobileIPProfile_t, 579
addressSize
unpack_sms_GetSMSCAddress_t, 521
aid
uim_refreshevent, 393
uim_sessionInformation, 395
uim_UIMSessionInformation, 399
aidLength
appStats, 44
uim_appStatus, 378
uim_refreshevent, 393
uim_sessionInformation, 395
uim_UIMSessionInformation, 399
aidVal
appStats, 44
uim_appStatus, 378
aidingIndicatorMask
loc_sensorDataUsage, 97
airTimerValue
voice_airTimer, 598
alertingPattern
voice_arrAlertingPattern, 601
AlertingType
voice_arrAlertingType, 602
alertmsg
unpack_omaDmConfigTlv_t, 494
alertmsglength
unpack_omaDmConfigTlv_t, 494
allCallsAlphaIDInfoArr
voice_arrAlphaID, 603
AllCallsUUSInfo
voice_arrUUSInfo, 608
alphaDcs
voice_alphaIDInfo, 601
AlphaIDInfo
voice_allCallsAlphaIDInfo, 599
alphaLen
voice_alphaIDInfo, 601
alphaText
voice_alphaIDInfo, 601
altSrcInfo_t, 40
coverage, 41
linkage, 41
source, 41
altitudeAssumed
unpack_loc_GnssSvInfo_Ind_t, 444
altitudeSrcInfo

pack_loc_SLQSLOCInjectPosition_t, 245
altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 245
altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 245
ambr_dl
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 505
ambr_dl_ext
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 505
ambr_dl_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 505
ambr_ul
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 505
ambr_ul_ext
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 505
ambr_ul_ext2
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 505
amssSize
unpack_dms_GetFirmwareRevision_t, 406
unpack_dms_GetFirmwareRevisions_t, 407
apdoxypages.c, 653
apnId
pack_qos_SLQSQosSwiReadApnExtraParams_t, 278
pack_qos_SLQSQosSwiReadDataStats_t, 278
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 505
unpack_qos_SLQSQosSwiReadDataStats_t, 507
apnName
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
apnname
unpack_wds_GetDefaultProfile_t, 577
apnsize
unpack_wds_GetDefaultProfile_t, 577
appNameLength
loc_LocApplicationInfo, 93
appProviderLength
loc_LocApplicationInfo, 93
appState
appStats, 44
uim_appStatus, 378
appStats, 41
aidLength, 44
aidVal, 44
appState, 44
appType, 44
persoFeature, 44
persoRetries, 44
persoState, 44
persoUnblockRetries, 44
pin1Retries, 44
pin1State, 44
pin2Retries, 44
pin2State, 44
puk1Retries, 44
puk2Retries, 44
univPin, 44
AppStatus
slotInf, 355
uim_slotInfo, 398
appType
appStats, 44
uim_appStatus, 378
appVersionLength
loc_LocApplicationInfo, 93
appVersionValid
loc_LocApplicationInfo, 93
Application
unpack_nas_GetCDMANetworkParameters_t, 454
appversion_str
unpack_dms_GetFirmwareInfo_t, 406
arfcn
nas_GERANInfo, 126
nas_gsmCellInfo, 127
arrfileInfo
uim_refreshevent, 393
uim_registerRefresh, 394
auth
unpack_wds_GetDefaultProfile_t, 577
authData
pack_uim_SLQSUIMAuthenticate_t, 301
AuthProt
nas_protocolSubtypeElement, 176
Authentication
unpack_wds_SLQSGetRuntimeSettings_t, 590
authentication
pack_wds_SetDefaultProfile_t, 332
AutoConnect
pack_swiavms_SLQSAvMSSetSettings_t, 291
unpack_swiavms_SLQSAvMSGetSettings_t, 530
AutoReboot
pack_swiavms_SLQSAvMSSetSettings_t, 291
unpack_swiavms_SLQSAvMSGetSettings_t, 530
Autosdm
unpack_swioma_SLQSOMADMGetSettings_t, 538
avgPeriod
nas_LTESigRptCfg, 151
nas_LTESigRptConfig, 152
azimuth
loc_satelliteInfo, 96

bAPNLength
_PackSwiAVMSSettingsAPNInfo, 29
_PackSwiAvmsSetSettingsAPNInfo, 28
bAuto
pack_pds_SetPortAutomaticTracking_t, 277
bBinaryType
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 33
bEnable

pack_dms_UIMSetPINProtection_t, 233
pack_nas_SLQSSetSignalStrengthsCallback_t, 270
bForceDownload
pack_fms_SetImagesPreference_t, 237
bICCID
slot_t, 353
bICCIDLength
slot_t, 353
bLogicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 307
slot_t, 353
bNotification
_UnpackSwiAvmsEventReportNotification, 38
bNumberOfPhySlots
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 542
BOOL
SwiDataTypes.h, 878
BPTlv
NASQmiCbkNasSystemSelPrefInd, 220
bPWDLength
_PackSwiAVMSSettingsAPNInfo, 29
_PackSwiAvmsSetSettingsAPNInfo, 28
bPackageID
_UnpackSwiAvmsEventReportPackageID, 39
bRegStatus
_UnpackSwiAvmsEventReportRegStatus, 39
bResetStatistics
rmTrasnferStaticsReq, 351
bSerity
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 33
bState
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 33
_UnpackSwiAvmsEventReportConfig, 34
bType
_UnpackSwiAvmsEventReportDataSessionStatus, 36
_UnpackSwiAvmsEventReportSessionType, 40
bUnameLength
_PackSwiAVMSSettingsAPNInfo, 29
_PackSwiAvmsSetSettingsAPNInfo, 28
bUserInputRequest
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 33
_UnpackSwiAvmsEventReportConfig, 35
_UnpackSwiAvmsEventReportConnectionRequest, 35
BYT_STAT_STAT_MASK
wds.h, 939
BYTE
SwiDataTypes.h, 878
band
nas_LTEInfo, 143
band1900
nas_gsmCellInfo, 128
band_pref
NASBandPreferenceTlv, 213
BandCapability
unpack_dms_GetBandCapability_t, 401
bandCapability
unpack_dms_SLQSGetBandCapability_t, 421
bandPref
pack_nas_SLQSSetBandPreference_t, 269
bandwidth
nas_LTEInfo, 143
baseId
nas_CDMAInfo, 106
nas_CDMASysInfo, 110
baseLat
nas_CDMAInfo, 106
nas_CDMASysInfo, 110
baseLong
nas_CDMAInfo, 106
nas_CDMASysInfo, 110
BasestationID
unpack_nas_SLQSGetServingSystem_t, 470
BasestationLatitude
unpack_nas_SLQSGetServingSystem_t, 470
BasestationLongitude
unpack_nas_SLQSGetServingSystem_t, 470
BearerID
unpack_qos_QosFlowInfo_t, 503
bearerID
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 592
bearerId
unpack_QosFlowStat_t, 519
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
bootSize
unpack_dms_GetFirmwareRevisions_t, 407
BootString
unpack_dms_GetFirmwareRevisions_t, 407
bootversion_str
unpack_dms_GetFirmwareInfo_t, 406
Broadcast
unpack_nas_GetCDMANetworkParameters_t, 454
broadcastActivate
pack_sms_SLQSSetSmsBroadcastActivation_t, 287
broadcastConfig
sms_qaQmi3GPPBroadcastCfgInfo, 363
bsInfoValid
nas_CDMASysInfo, 110
bsPRev
nas_CDMASysInfo, 110
bsPRevValid
nas_CDMASysInfo, 110
bsic
nas_GERANInfo, 126
bsicId
nas_gsmCellInfo, 128
bucketSz

unpack_qos_tokenBucket_t, 517
buildID
FMSImageIdElement, 55
image_info_t, 58
buildIDLen
image_info_t, 58
buildIDLength
FMSImageIdElement, 55
buildId
FMSImageElement, 54
buildIdLength
FMSImageElement, 54
BurstDTMFInfo
pack_voice_SLQSVoiceBurstDTMF_t, 313
ByteLoopbackMode
unpack_wds_SLQSSGetLoopback_t, 595
ByteLoopbackMultiplier
unpack_wds_SLQSSGetLoopback_t, 595

CDMA_P_Rev
unpack_nas_SLQSGetServingSystem_t, 470
CDMABroadcastConfig
sms_qaQmi3GPP2BroadcastCfgInfo, 362
CDMAECIOThreshListLen
nas_CDMAECIOThresh, 105
CDMARSSIThreshListLen
nas_CDMARSSIThresh, 107
CDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 481
CDMASystemInfoExt
unpack_nas_SLQSGetServingSystem_t, 470
CHAR
SwiDataTypes.h, 878
CK_MAX_SIZE
dms.h, 664
CQIValueCW0
unpack_nas_SLQSSwiGetLteCQI_t, 489
CQIValueCW1
unpack_nas_SLQSSwiGetLteCQI_t, 489
CSDomain
unpack_nas_GetServingNetwork_t, 460
CUGIndex
voice_CUGInfo, 622
CallBarStatus
unpack_nas_SLQSGetServingSystem_t, 470
callDuration
unpack_wds_GetSessionDuration_t, 583
callEndReason
unpack_wds_SLQSGetDUNCallInfo_t, 588
voice_arrCallEndReason, 604
CallFWExtInfo
voice_getCallFWExtInfo, 626
CallFWInfo
voice_getCallFWInfo, 627
callID
pack_voice_SLQSVoiceGetCallInfo_t, 317
pack_voice_SLQSVoiceStopContDTMF_t, 328
unpack_voice_SLQSVoiceStopContDTMF_t, 573
voice_allCallsAlphaIDInfo, 599
voice_allCallsDiagInfo, 599
voice_allCallsUUSInfo, 600
voice_arrAlertingPattern, 601
voice_arrAlertingType, 602
voice_arrCallEndReason, 604
voice_arrSvcOption, 608
voice_callInfo, 615
voice_getAllCallRmtPtyName, 625
voice_getAllCallRmtPtyNum, 626
voice_peerNumberInfo, 628
callNumber
pack_voice_SLQSVoiceDialCall_t, 314
callState
voice_callInfo, 615
callType
voice_callInfo, 615
CalledPartyNum
voice_arrCalledPartyNum, 603
callerID
voice_connectNumInfo, 621
callerIDLen
voice_connectNumInfo, 621
callerName
voice_remotePartyName, 631
Callinfo
voice_getAllCallInformation, 625
cardState
slotInf, 355
uim_slotInfo, 398
carrier
unpack_dms_SLQSSwiGetFirmwareCurr_t, 424
carrier_str
unpack_dms_GetFirmwareInfo_t, 406
CarrierImage_t, 44
m_FwBuildId, 45
m_FwImageId, 45
m_PriBuildId, 45
m_PriImageId, 45
m_nCarrierId, 45
m_nFolderId, 45
m_nStorage, 45
ccsSupported
nas_CDMASysInfo, 110
ccsSupportedValid
nas_CDMASysInfo, 110
cdmaSSInfo, 46
ecio, 46
rssi, 46
cdmaSysIdValid
nas_CDMASysInfo, 110
cell_resel_priority
nas_infoInterFreq, 138
cellBroadcastCap
nas_AddSysInfo, 102
CellID
unpack_nas_SLQSGetServingSystem_t, 470
cellID
nas_GERANInfo, 126

nas_UMTSInfo, 203
cellId
nas_GSMSysInfo, 131
nas_LTESysInfo, 156
nas_WCDMASysInfo, 212
cellIdValid
nas_gsmCellInfo, 128
nas_GSMSysInfo, 131
nas_LTESysInfo, 156
nas_WCDMASysInfo, 212
cellInterFreqParams
nas_infoInterFreq, 138
cellIsTDD
nas_umtsLTENbrCell, 205
CellParams
nas_LTEInfoIntrafreq, 146
cellReselPriority
nas_lteGsmCellInfo, 141
nas_LTEInfoIntrafreq, 146
nas_lteWcdmaCellInfo, 157
cells_len
nas_infoInterFreq, 138
nas_lteGsmCellInfo, 141
cellsLen
nas_LTEInfoIntrafreq, 146
nas_lteWcdmaCellInfo, 157
chaddr
wds_DHCPv4HWConfig, 638
wdsDhcpv4HwConfig, 649
chaddrLen
wds_DHCPv4HWConfig, 638
wdsDhcpv4HwConfig, 649
changePIN
pack_uim_ChangePin_t, 298
channelRate
unpack_wds_SLQSGetDUNCallInfo_t, 588
Chipset
nas_DeviceConfigDetail, 121
ckLen
uim_depersonalizationInformation, 384
ckVal
uim_depersonalizationInformation, 384
clear
pack_dms_SLQSSwiGetCrashInfo_t, 229
codingScheme
nas_PLMNNetworkNameData, 175
voice_remotePartyName, 631
common.h
eCAT, 656
eCTL, 656
eDMS, 656
eIND, 657
eLOC, 656
eLOG_DEBUG, 655
eLOG_FATAL, 656
eLOG_INFO, 655
eLOG_WARN, 655
eNAS, 656
ePDS, 656
eQOS, 656
eREQ, 657
eRMS, 656
eRSP, 657
eSAR, 656
eSMS, 656
eSWILOC, 656
eSWIM2MCMD, 656
eSWIOMA, 656
eSWIOMAEXT, 656
eTIMEOUT_10_S, 656
eTIMEOUT_20_S, 656
eTIMEOUT_2_S, 656
eTIMEOUT_300_S, 656
eTIMEOUT_30_S, 656
eTIMEOUT_5_S, 656
eTIMEOUT_60_S, 656
eTIMEOUT_8_S, 656
eTIMEOUT_DEFAULT, 656
eTMD, 656
eUIM, 656
eVOICE, 656
eWDS, 656
common.h, 653
eLOG_LEVEL, 655
eQMI_SVC, 656
eTimeout, 656
fill_pack_ctx, 657
fill_sdu_hdr, 657
get_version, 657
glog, 658
gloglvl, 658
helper_get_resp_ctx, 657
helper_get_xid, 657
helper_set_log_func, 657
helper_set_log_lvl, 657
liteqmi_GetVersion, 657
liteqmi_log, 657
logger, 655
MINREQBKLEN, 655
MSGID_AND_LEN, 655
MSGID_DONT_CARE, 655
msgtype, 656
SDU_HDR_LEN, 655
UNUSEDPARAM, 655
unpack_result_code_only, 658
commonInfo
unpack_nas_SLQSNasSwiModemStatus_t, 485
ConcSvcInfo
unpack_nas_SLQSGetServingSystem_t, 470
confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 249
conn_status
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 592
ConnectedPartyNum
voice_arrConnectPartyNum, 605

connectionStatus, 46
MDMCallDuration, 46
MDMConnStatus, 46
unpack_wds_GetSessionState_t, 583
unpack_wds_SLQSGetDUNCallInfo_t, 588
content
uim_authenticateResult, 379
uim_readResult, 391
contentLen
uim_authenticateResult, 379
uim_readResult, 391
context
uim_authenticationData, 380
contextId
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 348
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
contextType
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 348
ControlMac
nas_protocolSubtypeElement, 176
countryInitials
nas_PLMNNetworkNameData, 175
coverage
altSrcInfo_t, 41
cpich_ecno
nas_wcdmaCellInfo, 207
cpich_rscp
nas_wcdmaCellInfo, 207
crashAction
pack_dms_SetCrashAction_t, 226
crashData
crashInformation, 48
crashId
crashInformation, 48
crashInfo
crashInfoParams, 47
crashInfoParam
unpack_dms_SLQSSwiGetCrashInfo_t, 422
crashInfoParams, 46
crashInfo, 47
crashStatus, 47
crashInformation, 47
crashData, 48
crashId, 48
crashString, 48
crashStrlen, 48
gcdumpString, 48
gcdumpStrlen, 48
numCrashes, 48
crashStatus
crashInfoParams, 47
crashString
crashInformation, 48
crashStrlen
crashInformation, 48
csAttachState
nas_servSystem, 190
NASServingSystemInfo, 222
csBarStatus
nas_CallBarringSysInfo, 103
nas_callBarStatus, 104
cur_carr_name
unpack_dms_GetFirmwareInfo_t, 406
cur_carr_rev
unpack_dms_GetFirmwareInfo_t, 406
curDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 587
curProfile
pack_wds_SLQSModifyProfile_t, 340
UnPackGetProfileSettingOut, 598
CurrChanRxRate
dunchannelRate, 53
CurrChanTxRate
dunchannelRate, 53
currDBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 593
currNWInfo
unpack_wds_SLQSSetWdsEventCallback_ind_t, 593
currNetworkInfo, 48
NetworkType, 48
RATMask, 48
SOMask, 48
unpack_wds_SLQSGetCurrDataSystemStat_t, 586
current_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 587
current_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 587
currentChannelRXRate
unpack_wds_GetConnectionRate_t, 575
currentChannelTXRate
unpack_wds_GetConnectionRate_t, 575
currentDataBearer
pack_wds_SLQSSetWdsEventCallback_t, 342
currentNetwork
qmiWDSDataBearerTechnology, 350
CurrentPLMN
unpack_nas_SLQSGetServingSystem_t, 470
cust_attr
DMScustSettingInfo, 51
cust_id
DMScustSettingInfo, 51
DMSgetCustomInput, 52
pack_dms_GetCustFeaturesV2_t, 224
pack_dms_SetCustFeaturesV2_t, 227
cust_value
DMScustSettingInfo, 51
pack_dms_SetCustFeaturesV2_t, 227

custSetting
DMScustSettingList, 51
CustomSCP
unpack_nas_GetCDMANetworkParameters_t, 455

dBTechAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 593
dBTechExtRatValue
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
dBTechExtSoMask
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
dBTechnology
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
dBtechExtAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
dBtechnologyExt
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
DHCPRelayEnabled
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
DHCPv4LeaseOptTlv
unpack_wds_DHCPv4ClientLease_ind_t, 574
DHCPv4LeaseStateTlv
unpack_wds_DHCPv4ClientLease_ind_t, 574
DMS_IMGDETAILS_LEN
dms.h, 664
DMS_PM_FACTORY
dms.h, 664
DMS_PM_LOW
dms.h, 664
DMS_PM_OFFLINE
dms.h, 664
DMS_PM_ONLINE
dms.h, 664
DMS_PM_RESET
dms.h, 664
DMS_PM_SHUT_DOWN
dms.h, 664
DMScustSettingInfo, 50
cust_attr, 51
cust_id, 51
cust_value, 51
id_length, 51
value_length, 51
DMScustSettingList, 51
custSetting, 51
list_type, 51
num_instances, 51
DMSgetCustomFeatureV2, 51
pCustSettingInfo, 52
pCustSettingList, 52
pGetCustomInput, 52
DMSgetCustomInput, 52
cust_id, 52
list_type, 52
DRCCover
nas_DRCParams, 122
DRCValue
nas_DRCParams, 122
DTMFInterdigitInterval
voice_DTMFLengths, 624
DTMFPulseWidth
voice_DTMFLengths, 624
DTMFdigit
pack_voice_SLQSVoiceStartContDTMF_t, 327
DTMInd
unpack_nas_SLQSGetServingSystem_t, 470
data
sMSCAddress, 369
sMSEtwsMessage, 370
sMSTransferRouteMTMessage, 373
uim_authenticationData, 380
data_buf
NASOTAMessageTlv, 216
data_len
NASOTAMessageTlv, 216
dataBearer
pack_wds_SLQSSetWdsEventCallback_t, 342
dataBearerMask
unpack_wds_SLQSGetDataBearerTechnology_t, 587
dataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 588
dataBearerTechExt
pack_wds_SLQSSetWdsEventCallback_t, 343
dataCapabilities
nas_dataSrvCapabilities, 116
dataCapabilitiesLen
nas_dataSrvCapabilities, 116
DataCaps
unpack_nas_GetServingNetwork_t, 460
unpack_nas_GetServingNetworkCapabilities_t, 460
dataCaps
unpack_nas_SetDataCapabilitiesCallback_ind_t, 462
DataCapsLen
unpack_nas_GetServingNetwork_t, 460
unpack_nas_GetServingNetworkCapabilities_t, 460
dataCapsSize
unpack_nas_SetDataCapabilitiesCallback_ind_t, 462
dataLen
uim_authenticationData, 380
DataRate
unpack_qos_swiQosFlow_t, 516
dataRateMax
unpack_qos_dataRate_t, 499
dataServiceCaCapability
unpack_dms_GetDeviceCapabilities_t, 404

DataServiceCapability
unpack_dms_GetDeviceCap_t, 403
DataSrvCapabilities
unpack_nas_SLQSGetServingSystem_t, 470
dataSysStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 593
dataSystemStatus
pack_wds_SLQSSetWdsEventCallback_t, 343
Date
unpack_swioma_SLQSOMADMGetSessionInfo_t, 536
DateLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 536
day
nas_timeInfo, 200
nas_UniversalTime, 206
dayLtSavingAdj
nas_timeInfo, 200
dayOfWeek
nas_timeInfo, 200
nas_UniversalTime, 206
daylightSavings
nas_qaQmi3Gpp2TimeZone, 177
defaultPDNEnabled
unpack_wds_SLQSGet3GPPConfigItem_t, 585
DefaultRoamInd
unpack_nas_SLQSGetServingSystem_t, 470
delayClass
LibPackGPRSRequestedQoS, 59
wds_GPRSQoS, 642
deliveryErrSDU
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 648
depersonilisationInfo
pack_uim_SLQSUIMDepersonalization_t, 302
description
unpack_omaDmFotaTlv_t, 496
descriptionlength
unpack_omaDmFotaTlv_t, 496
Desription
nas_QmiNas3GppNetworkInfo, 178
destPortRangeEnd
LibPackTFTIDParams, 84
destPortRangeStart
LibPackTFTIDParams, 84
DetailedSvcInfo
unpack_nas_SLQSGetServingSystem_t, 470
DevCrashState
unpack_dms_GetCrashAction_t, 401
DiagInfo
voice_allCallsDiagInfo, 599
voice_arrDiagInfo, 605
diagInfoLen
voice_diagInfo, 623
diagnosticInfo
voice_diagInfo, 623
digitCnt
voice_burstDTMFInfo, 609
dirNum
nas_dirNum, 121
dirNumLen
nas_dirNum, 121
direction
voice_callInfo, 615
DisableIMSI
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
displayCondition
nas_serviceProviderName, 188
dl_bw_value
nas_PhyCaAggPcellInfo, 167
nas_PhyCaAggScellDlBw, 168
nas_PhyCaAggScellInfo, 172
NASPhyCaAggPcellInfo, 217
NASPhyCaAggScellDlBw, 217
NASPhyCaAggScellInfo, 219
dms.h, 658
ACT_CODE_MAX_SIZE, 664
CK_MAX_SIZE, 664
DMS_IMGDETAILS_LEN, 664
DMS_PM_FACTORY, 664
DMS_PM_LOW, 664
DMS_PM_OFFLINE, 664
DMS_PM_ONLINE, 664
DMS_PM_RESET, 664
DMS_PM_SHUT_DOWN, 664
ERI_DATA_MAX_SIZE, 665
MAX_BUILD_ID_LEN, 665
MEID_MAX_SIZE, 665
pack_dms_ActivateAutomatic, 665
pack_dms_GetActivationState, 666
pack_dms_GetBandCapability, 666
pack_dms_GetCrashAction, 666
pack_dms_GetCustFeature, 667
pack_dms_GetCustFeaturesV2, 667
pack_dms_GetDeviceCap, 667
pack_dms_GetDeviceCapabilities, 667
pack_dms_GetDeviceHardwareRev, 668
pack_dms_GetDeviceMfr, 668
pack_dms_GetDeviceSerialNumbers, 668
pack_dms_GetFSN, 670
pack_dms_GetFirmwareInfo, 669
pack_dms_GetFirmwareRevision, 669
pack_dms_GetFirmwareRevisions, 670
pack_dms_GetHardwareRevision, 670
pack_dms_GetIMSI, 671
pack_dms_GetManufacturer, 671
pack_dms_GetModelID, 671
pack_dms_GetNetworkTime, 672
pack_dms_GetOfflineReason, 672
pack_dms_GetPRLVersion, 673
pack_dms_GetPower, 672
pack_dms_GetSerialNumbers, 673
pack_dms_GetUSBComp, 673

pack_dms_GetVoiceNumber, 674
pack_dms_ResetToFactoryDefaults, 674
pack_dms_SLQSDmsSwiGetResetInfo, 677
pack_dms_SLQSDmsSwiIndicationRegister, 678
pack_dms_SLQSGetBandCapability, 678
pack_dms_SLQSGetERIFile, 679
pack_dms_SLQSSwiClearDyingGaspStatistics, 679
pack_dms_SLQSSwiGetCrashInfo, 679
pack_dms_SLQSSwiGetDyingGaspCfg, 680
pack_dms_SLQSSwiGetDyingGaspStatistics, 680
pack_dms_SLQSSwiGetFirmwareCurr, 680
pack_dms_SLQSSwiGetFwUpdateStatus, 681
pack_dms_SLQSSwiGetHostDevInfo, 681
pack_dms_SLQSSwiGetOSInfo, 681
pack_dms_SLQSSwiGetSerialNoExt, 682
pack_dms_SLQSSwiSetDyingGaspCfg, 682
pack_dms_SLQSSwiSetHostDevInfo, 682
pack_dms_SLQSSwiSetOSInfo, 683
pack_dms_SLQSUIMGetState, 683
pack_dms_SetActivationStatusCallback, 675
pack_dms_SetCrashAction, 675
pack_dms_SetCustFeature, 675
pack_dms_SetCustFeaturesV2, 676
pack_dms_SetEventReport, 676
pack_dms_SetFirmwarePreference, 676
pack_dms_SetPower, 677
pack_dms_SetUSBComp, 677
pack_dms_UIMChangePIN, 683
pack_dms_UIMGetControlKeyStatus, 684
pack_dms_UIMGetICCID, 684
pack_dms_UIMGetPINStatus, 685
pack_dms_UIMSetControlKeyProtection, 685
pack_dms_UIMSetPINProtection, 685
pack_dms_UIMUnblockControlKey, 686
pack_dms_UIMUnblockPIN, 686
pack_dms_UIMVerifyPIN, 686
pack_dms_ValidateSPC, 687
SPC_SIZE, 665
UNIQUE_ID_LEN, 665
unpack_dms_ActivateAutomatic, 687
unpack_dms_GetActivationState, 687
unpack_dms_GetBandCapability, 688
unpack_dms_GetCrashAction, 688
unpack_dms_GetCustFeature, 689
unpack_dms_GetCustFeaturesV2, 689
unpack_dms_GetDeviceCap, 689
unpack_dms_GetDeviceCapabilities, 689
unpack_dms_GetDeviceHardwareRev, 690
unpack_dms_GetDeviceMfr, 690
unpack_dms_GetDeviceSerialNumbers, 690
unpack_dms_GetFSN, 692
unpack_dms_GetFirmwareInfo, 691
unpack_dms_GetFirmwareRevision, 691
unpack_dms_GetFirmwareRevisions, 691
unpack_dms_GetHardwareRevision, 692
unpack_dms_GetIMSI, 692
unpack_dms_GetManufacturer, 693
unpack_dms_GetModelID, 693
unpack_dms_GetNetworkTime, 693
unpack_dms_GetOfflineReason, 694
unpack_dms_GetPRLVersion, 694
unpack_dms_GetPower, 694
unpack_dms_GetSerialNumbers, 695
unpack_dms_GetUSBComp, 695
unpack_dms_GetVoiceNumber, 695
unpack_dms_ResetToFactoryDefaults, 696
unpack_dms_SLQSDmsSwiGetResetInfo, 699
unpack_dms_SLQSDmsSwiGetResetInfo_Ind, 699
unpack_dms_SLQSDmsSwiIndicationRegister, 700
unpack_dms_SLQSGetBandCapability, 700
unpack_dms_SLQSGetERIFile, 701
unpack_dms_SLQSSwiClearDyingGaspStatistics, 701
unpack_dms_SLQSSwiGetCrashInfo, 701
unpack_dms_SLQSSwiGetDyingGaspCfg, 702
unpack_dms_SLQSSwiGetDyingGaspStatistics, 702
unpack_dms_SLQSSwiGetFirmwareCurr, 702
unpack_dms_SLQSSwiGetFwUpdateStatus, 703
unpack_dms_SLQSSwiGetHostDevInfo, 703
unpack_dms_SLQSSwiGetOSInfo, 703
unpack_dms_SLQSSwiGetSerialNoExt, 704
unpack_dms_SLQSSwiSetDyingGaspCfg, 704
unpack_dms_SLQSSwiSetHostDevInfo, 704
unpack_dms_SLQSSwiSetOSInfo, 705
unpack_dms_SLQSUIMGetState, 705
unpack_dms_SetActivationStatusCallback, 696
unpack_dms_SetCrashAction, 696
unpack_dms_SetCustFeature, 697
unpack_dms_SetCustFeaturesV2, 697
unpack_dms_SetEventReport, 697
unpack_dms_SetEventReport_ind, 698
unpack_dms_SetFirmwarePreference, 698
unpack_dms_SetPower, 698
unpack_dms_SetUSBComp, 699
unpack_dms_UIMChangePIN, 705
unpack_dms_UIMGetControlKeyStatus, 706
unpack_dms_UIMGetICCID, 706
unpack_dms_UIMGetPINStatus, 706
unpack_dms_UIMSetControlKeyProtection, 707
unpack_dms_UIMSetPINProtection, 707
unpack_dms_UIMUnblockControlKey, 707
unpack_dms_UIMUnblockPIN, 708
unpack_dms_UIMVerifyPIN, 708
unpack_dms_ValidateSPC, 708
dms_ActivationStatusTlv, 48
activationStatus, 49
TlvPresent, 49
dms_OperatingModeTlv, 49
operatingMode, 50
TlvPresent, 50
domain
wds_DomainNameList, 641

domainLen
wds_Domain, 641
DomainList
unpack_wds_SLQSGetRuntimeSettings_t, 590
domainName
wds_Domain, 641
dormancyStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
dormancyState
unpack_wds_GetDormancyState_t, 578
dormancyStatus
pack_wds_SLQSSetWdsEventCallback_t, 343
unpack_wds_SLQSGetDUNCallInfo_t, 588
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
dtmSupp
nas_GSMSysInfo, 131
dtmSuppValid
nas_GSMSysInfo, 131
dunchannelRate, 52
CurrChanRxRate, 53
CurrChanTxRate, 53
MaxChanRxRate, 53
MaxChanTxRate, 53
Duration
pack_nas_SetNetworkPreference_t, 254
unpack_nas_GetNetworkPreference_t, 458

eCAT
common.h, 656
eCTL
common.h, 656
eDMS
common.h, 656
eFW_TYPE_CWE
lite-fw.h, 715
eFW_TYPE_CWE_NVU
lite-fw.h, 715
eFW_TYPE_INVALID
lite-fw.h, 715
eFW_TYPE_MBN
lite-fw.h, 715
eFW_TYPE_MBN_GOBI
lite-fw.h, 715
eFW_TYPE_NVU
lite-fw.h, 715
eFW_TYPE_SPK
lite-fw.h, 715
eIND
common.h, 657
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100
nas.h, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15
nas.h, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25
nas.h, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50
nas.h, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6
nas.h, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75
nas.h, 742
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED
nas.h, 743
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED
nas.h, 743
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED
nas.h, 743
eLOC
common.h, 656
eLOG_DEBUG
common.h, 655
eLOG_FATAL
common.h, 656
eLOG_INFO
common.h, 655
eLOG_WARN
common.h, 655
eModel_9X15
lite-fw.h, 716
eModel_9X30
lite-fw.h, 716
eModel_Unknown
lite-fw.h, 716
eNAS
common.h, 656
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100
nas.h, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15
nas.h, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25
nas.h, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50
nas.h, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6
nas.h, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75
nas.h, 743
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE
nas.h, 743
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE
nas.h, 743
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE
nas.h, 743
ePDS
common.h, 656
eQCWWAN_ERR_API_MUTEX_TIMEOUT
qmerrno.h, 821
eQCWWAN_ERR_BUFFER_SZ
qmerrno.h, 820

eQCWWAN_ERR_CANCEL_OP
qmerrno.h, 821
eQCWWAN_ERR_DRIVER
qmerrno.h, 821
eQCWWAN_ERR_ENUM_BEGIN
qmerrno.h, 820
eQCWWAN_ERR_ENUM_END
qmerrno.h, 821
eQCWWAN_ERR_FILE_COPY
qmerrno.h, 820
eQCWWAN_ERR_FILE_OPEN
qmerrno.h, 820
eQCWWAN_ERR_GENERAL
qmerrno.h, 820
eQCWWAN_ERR_INTERNAL
qmerrno.h, 820
eQCWWAN_ERR_INVALID_ARG
qmerrno.h, 820
eQCWWAN_ERR_INVALID_DEVID
qmerrno.h, 820
eQCWWAN_ERR_INVALID_FILE
qmerrno.h, 820
eQCWWAN_ERR_INVALID_QMI_RSP
qmerrno.h, 820
eQCWWAN_ERR_INVALID_XID
qmerrno.h, 821
eQCWWAN_ERR_MALFORMED_QMI_RSP
qmerrno.h, 820
eQCWWAN_ERR_MEMORY
qmerrno.h, 820
eQCWWAN_ERR_MULTIPLE_DEVICES
qmerrno.h, 821
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED
qmerrno.h, 821
eQCWWAN_ERR_NO_CANCELABLE_OP
qmerrno.h, 821
eQCWWAN_ERR_NO_CONNECTION
qmerrno.h, 820
eQCWWAN_ERR_NO_DEVICE
qmerrno.h, 820
eQCWWAN_ERR_NO_SIGNAL
qmerrno.h, 821
eQCWWAN_ERR_NONE
qmerrno.h, 820
eQCWWAN_ERR_NULL_TLV
qmerrno.h, 824
eQCWWAN_ERR_OFFLINE
qmerrno.h, 821
eQCWWAN_ERR_PDU_GENERATION
qmerrno.h, 821
eQCWWAN_ERR_QMI_ABORTED
qmerrno.h, 821
eQCWWAN_ERR_QMI_ACCESS_DENIED
qmerrno.h, 823
eQCWWAN_ERR_QMI_ACK_NOT_SENT
qmerrno.h, 823
eQCWWAN_ERR_QMI_ARG_TOO_LONG
qmerrno.h, 821
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED
qmerrno.h, 822
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK
qmerrno.h, 822
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED
qmerrno.h, 823
eQCWWAN_ERR_QMI_CALL_FAILED
qmerrno.h, 821
eQCWWAN_ERR_QMI_CARD_BUSY_RSP
qmerrno.h, 824
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED
qmerrno.h, 823
eQCWWAN_ERR_QMI_CAT_END
qmerrno.h, 824
eQCWWAN_ERR_QMI_CAT_START
qmerrno.h, 824
eQCWWAN_ERR_QMI_CAUSE_CODE
qmerrno.h, 822
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED
qmerrno.h, 821
eQCWWAN_ERR_QMI_CONNECT
qmerrno.h, 820
eQCWWAN_ERR_QMI_DEVICE_IN_USE
qmerrno.h, 821
eQCWWAN_ERR_QMI_DEVICE_NOT_READY
qmerrno.h, 822
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL
qmerrno.h, 822
eQCWWAN_ERR_QMI_DISABLED
qmerrno.h, 822
eQCWWAN_ERR_QMI_ENCODING
qmerrno.h, 822
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE
qmerrno.h, 824
eQCWWAN_ERR_QMI_EVENT_REG_FAILED
qmerrno.h, 824
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL
qmerrno.h, 823
eQCWWAN_ERR_QMI_FDN_RESTRICT
qmerrno.h, 823
eQCWWAN_ERR_QMI_FLOW_SUSPENDED
qmerrno.h, 822
eQCWWAN_ERR_QMI_GENERAL
qmerrno.h, 822
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED
qmerrno.h, 823
eQCWWAN_ERR_QMI_IFACE
qmerrno.h, 820
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE
qmerrno.h, 823
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER
qmerrno.h, 822
eQCWWAN_ERR_QMI_INCORRECT_PIN
qmerrno.h, 821
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE
qmerrno.h, 823

eQCWWAN_ERR_QMI_INJECT_TIMEOUT
qmerrno.h, 823
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES
qmerrno.h, 822
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND
qmerrno.h, 822
eQCWWAN_ERR_QMI_INTERNAL
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_ARG
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD
qmerrno.h, 824
eQCWWAN_ERR_QMI_INVALID_HANDLE
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_ID
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_INDEX
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_OPERATION
qmerrno.h, 823
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_PINID
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_PROFILE
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_QMI_CMD
qmerrno.h, 823
eQCWWAN_ERR_QMI_INVALID_QOS_ID
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_TECH_PREF
qmerrno.h, 821
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP
qmerrno.h, 824
eQCWWAN_ERR_QMI_INVALID_TRANSITION
qmerrno.h, 822
eQCWWAN_ERR_QMI_INVALID_TX_ID
qmerrno.h, 821
eQCWWAN_ERR_QMI_MALFORMED_MSG
qmerrno.h, 821
eQCWWAN_ERR_QMI_MAX
qmerrno.h, 823
eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE
qmerrno.h, 822
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE
qmerrno.h, 822
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE
qmerrno.h, 822
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT
qmerrno.h, 822
eQCWWAN_ERR_QMI_MISSING_ARG
qmerrno.h, 821
eQCWWAN_ERR_QMI_MSG_BLOCKED
qmerrno.h, 823
eQCWWAN_ERR_QMI_NETWORK_ABORTED
qmerrno.h, 823
eQCWWAN_ERR_QMI_NETWORK_NOT_READY
qmerrno.h, 822
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE
qmerrno.h, 822
eQCWWAN_ERR_QMI_NO_EFFECT
qmerrno.h, 821
eQCWWAN_ERR_QMI_NO_ENTRY
qmerrno.h, 822
eQCWWAN_ERR_QMI_NO_FREE_PROFILE
qmerrno.h, 821
eQCWWAN_ERR_QMI_NO_MEMORY
qmerrno.h, 821
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND
qmerrno.h, 821
eQCWWAN_ERR_QMI_NO_RADIO
qmerrno.h, 823
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION
qmerrno.h, 823
eQCWWAN_ERR_QMI_NO_THRESHOLDS
qmerrno.h, 821
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE
qmerrno.h, 822
eQCWWAN_ERR_QMI_NOT_PROVISIONED
qmerrno.h, 821
eQCWWAN_ERR_QMI_NOT_SUPPORTED
qmerrno.h, 823
eQCWWAN_ERR_QMI_OFFSET
qmerrno.h, 821
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED
qmerrno.h, 821
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED
qmerrno.h, 821
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE
qmerrno.h, 823

eQCWWAN_ERR_QMI_OUT_OF_CALL
qmerrno.h, 821
eQCWWAN_ERR_QMI_PIN_BLOCKED
qmerrno.h, 822
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED
qmerrno.h, 822
eQCWWAN_ERR_QMI_POLICY_MISMATCH
qmerrno.h, 823
eQCWWAN_ERR_QMI_REQ
qmerrno.h, 820
eQCWWAN_ERR_QMI_REQ_SCH
qmerrno.h, 820
eQCWWAN_ERR_QMI_REQ_TO
qmerrno.h, 820
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED
qmerrno.h, 822
eQCWWAN_ERR_QMI_RSP
qmerrno.h, 820
eQCWWAN_ERR_QMI_RSP_TO
qmerrno.h, 820
eQCWWAN_ERR_QMI_SEGMENT_ORDER
qmerrno.h, 823
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG
qmerrno.h, 823
eQCWWAN_ERR_QMI_SESSION_INACTIVE
qmerrno.h, 822
eQCWWAN_ERR_QMI_SESSION_INVALID
qmerrno.h, 822
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP
qmerrno.h, 822
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND
qmerrno.h, 823
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED
qmerrno.h, 822
eQCWWAN_ERR_QMI_SMSC_ADDR
qmerrno.h, 823
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE
qmerrno.h, 823
eQCWWAN_ERR_QMI_TPDU_TYPE
qmerrno.h, 823
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION
qmerrno.h, 821
eQCWWAN_ERR_QMI_UNKNOWN
qmerrno.h, 822
eQCWWAN_ERR_QMI_WIDTH
qmerrno.h, 824
eQCWWAN_ERR_RESET
qmerrno.h, 821
eQCWWAN_ERR_SWICM_AM_VERS_ERROR
qmerrno.h, 823
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS
qmerrno.h, 823
eQCWWAN_ERR_SWICM_END
qmerrno.h, 824
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS
qmerrno.h, 823
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID
qmerrno.h, 823
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID
qmerrno.h, 823
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID
qmerrno.h, 823
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED
qmerrno.h, 823
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED
qmerrno.h, 823
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED
qmerrno.h, 823
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS
qmerrno.h, 823
eQCWWAN_ERR_SWICM_SOCKET_IN_USE
qmerrno.h, 823
eQCWWAN_ERR_SWICM_START
qmerrno.h, 823
eQCWWAN_ERR_SWICM_TIMEOUT
qmerrno.h, 823
eQCWWAN_ERR_SWICM_V4DWN_V6DWN
qmerrno.h, 823
eQCWWAN_ERR_SWICM_V4DWN_V6UP
qmerrno.h, 823
eQCWWAN_ERR_SWICM_V4UP_V6DWN
qmerrno.h, 823
eQCWWAN_ERR_SWICM_V4UP_V6UP
qmerrno.h, 823
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED
qmerrno.h, 824
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND
qmerrno.h, 824
eQCWWAN_ERR_SWIDCS_END
qmerrno.h, 824
eQCWWAN_ERR_SWIDCS_FILEIO_ERR
qmerrno.h, 824
eQCWWAN_ERR_SWIDCS_IOCTL_ERR
qmerrno.h, 824
eQCWWAN_ERR_SWIDCS_START
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_END
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE

qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_INVALID_PATH
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_OPENING_DIR
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_OPENING_FILE
qmerrno.h, 824
eQCWWAN_ERR_SWIIM_START
qmerrno.h, 824
eQCWWAN_ERR_SWISM_END
qmerrno.h, 824
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND
qmerrno.h, 824
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED
qmerrno.h, 824
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG
qmerrno.h, 824
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED
qmerrno.h, 824
eQCWWAN_ERR_SWISMS_START
qmerrno.h, 824
eQMI_LOC_SESS_STATUS_FAILURE
loc.h, 725
eQMI_LOC_SESS_STATUS_IN_PROGRESS
loc.h, 724
eQMI_LOC_SESS_STATUS_SUCCESS
loc.h, 724
eQMI_LOC_SESS_STATUS_TIMEOUT
loc.h, 725
eQOS
common.h, 656
eREQ
common.h, 657
eRMS
common.h, 656
eRSP
common.h, 657
eSAR
common.h, 656
eSDP_FWDWL_ERR_END
lite-fw.h, 716
eSDP_FWDWL_ERR_FAIL
lite-fw.h, 716
eSDP_FWDWL_ERR_FW_UPGRADE
lite-fw.h, 715
eSDP_FWDWL_ERR_FW_VERSION_FAIL
lite-fw.h, 716
eSDP_FWDWL_ERR_GENERAL
lite-fw.h, 715
eSDP_FWDWL_ERR_INVALID_DEV
lite-fw.h, 716
eSDP_FWDWL_ERR_INVALID_PATH
lite-fw.h, 716
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED
lite-fw.h, 715
eSDP_FWDWL_ERR_PATH_TOO_LONG
lite-fw.h, 715
eSDP_FWDWL_ERR_PRI_FAIL
lite-fw.h, 716
eSDP_FWDWL_ERR_SDK
lite-fw.h, 715
eSDP_FWDWL_ERR_SET_CBK
lite-fw.h, 715
eSDP_FWDWL_ERR_TIMEOUT
lite-fw.h, 716
eSDP_FWDWL_SUCCESS
lite-fw.h, 715
eSMS
common.h, 656
eSWILOC
common.h, 656
eSWIM2MCMD
common.h, 656
eSWIOMA
common.h, 656
eSWIOMAEXT
common.h, 656
eTIMEOUT_10_S
common.h, 656
eTIMEOUT_20_S
common.h, 656
eTIMEOUT_2_S
common.h, 656
eTIMEOUT_300_S
common.h, 656
eTIMEOUT_30_S
common.h, 656
eTIMEOUT_5_S
common.h, 656
eTIMEOUT_60_S
common.h, 656
eTIMEOUT_8_S
common.h, 656
eTIMEOUT_DEFAULT
common.h, 656

eTMD
common.h, 656
eUIM
common.h, 656
eVOICE
common.h, 656
eWDS
common.h, 656
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_END
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_FAIL
qmerrno.h, 825
eWDS_ERR_PROFILE_REG_RESULT_LIST_END
qmerrno.h, 825
ECIOThresListLen
nas_ECIOThresh, 123
eLOG_LEVEL
common.h, 655
EMTlv
NASQmiCbkNasSystemSelPrefInd, 220
eQCWWANError
qmerrno.h, 820
eQMI_SVC
common.h, 656
ERI_DATA_MAX_SIZE
dms.h, 665
ESNString
unpack_dms_GetDeviceSerialNumbers_t, 405
ETWSPLMNInfo
eTWSPLMNInfoTlv, 54
eTWSPLMNInfoTlv, 53
ETWSPLMNInfo, 54
TlvPresent, 54
ETWSPLMNTlv
unpack_sms_SetNewSMSCallback_ind_t, 523
ETWSTlv
unpack_sms_SetNewSMSCallback_ind_t, 523
eTimeout
common.h, 656
eValid
LibPackTFTIDParams, 84
earfcn
nas_infoInterFreq, 138
nas_LTEInfoIntrafreq, 146
nas_ltePCI, 148
nas_umtsLTENbrCell, 205
earfcn0
nas_lteEARFCN, 140
earfcn1
nas_lteEARFCN, 140
ecio
cdmaSSInfo, 46
hdrSSInfo, 58
nas_ecioListElement, 122
nas_rxInfo, 185
nas_UMTSInfo, 203
tdscdmaSigInfoExt, 374
ecioDelta
nas_SLQSSignalStrengthsIndReq, 191
ecioInfo
nas_SLQSSignalStrengthsInformation, 192
ecioList
unpack_nas_SLQSGetSignalStrength_t, 472
ecioListLen
unpack_nas_SLQSGetSignalStrength_t, 472
ecioThresholdList
nas_SLQSSignalStrengthsIndReq, 191

ecioThresholdListLen
nas_SLQSSignalStrengthsIndReq, 192
egprsSupp
nas_GSMSysInfo, 132
egprsSuppValid
nas_GSMSysInfo, 132
elevation
loc_satelliteInfo, 96
EmerMode
NASEmergencyModeTlv, 214
emmConnState
nas_LTEInfo, 143
emmState
nas_LTEInfo, 143
emmSubState
nas_LTEInfo, 143
enable
pack_pds_SetPDSState_t, 276
pack_qos_SLQSSetQosEventCallback_t, 279
enabled
pack_rms_SetSMSWake_t, 279
unpack_rms_GetSMSWake_t, 519
unpack_wds_GetMobileIPProfile_t, 579
EncryptProt
nas_protocolSubtypeElement, 176
EncryptedPIN1
pack_uim_ChangePin_t, 298
pack_uim_SetPinProtection_t, 300
pack_uim_UnblockPin_t, 308
engineState
unpack_loc_EngineState_Ind_t, 443
eqmiCbkSetStatus
sms.h, 842
eriData
eriDataparams, 53
eriDataLen
eriDataparams, 53
eriDataparams, 53
eriData, 53
eriDataLen, 53
eriFile
unpack_dms_SLQSGetERIFile_t, 421
error
unpack_wds_GetLastMobileIPError_t, 578
errorRate
nas_errorRateListElement, 124
errorRateInfo
nas_SLQSSignalStrengthsInformation, 192
errorRateList
unpack_nas_SLQSGetSignalStrength_t, 472
errorRateListLen
unpack_nas_SLQSGetSignalStrength_t, 472
errorState
slotInf, 355
uim_slotInfo, 398
esn
unpack_dms_GetSerialNumbers_t, 411
esnSize
unpack_dms_GetDeviceSerialNumbers_t, 405
EspSpi
unpack_qos_swiQosFilter_t, 512
EtwsMessageInfo
sMSEtwsMessageTlv, 370
event
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 508
unpack_qos_SLQSSetQosStatusCallback_ind_t, 509
eventMask
pack_uim_SLQSUIMEventRegister_t, 302
unpack_uim_SLQSUIMEventRegister_t, 544
eventRegister
pack_loc_EventRegister_t, 240
eventType
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 534
evrcCapability
voice_prefVoiceSO, 630
executingImage
FMSImageIDEntries, 56
exponent
unpack_qos_pktErrRate_t, 501
extBit
voice_calledPartySubAdd, 610
ExtErrorCode
PackCreateProfileOut, 348
extPowerState
pack_loc_SetExtPowerState_t, 240
extendedErrorCode
unpack_wds_SLQSDeleteProfile_t, 584

FLOAT
SwiDataTypes.h, 878
FMSImageElement, 54
buildId, 54
buildIdLength, 54
imageId, 54
imageType, 54
FMSImageIDEntries, 55
executingImage, 56
imageIDElement, 56
imageIDSize, 56
imageType, 56
maxImages, 56
FMSImageIdElement, 55
buildID, 55
buildIDLength, 55
failureCount, 55
imageID, 55
storageIndex, 55
FMSImageList, 56
imageIDEntries, 57
listSize, 57
FMSPrefImageList, 57
listEntries, 57
listSize, 57
FOTAUpdate

pack_swioma_SLQSOMADMSetSettings_t, 297
unpack_swioma_SLQSOMADMGetSettings_t, 538
FOTAdownload
pack_swioma_SLQSOMADMSetSettings_t, 297
unpack_swioma_SLQSOMADMGetSettings_t, 538
facility
pack_dms_UIMGetControlKeyStatus_t, 232
pack_dms_UIMSetControlKeyProtection_t, 233
pack_dms_UIMUnblockControlKey_t, 234
facilityCk
pack_dms_UIMSetControlKeyProtection_t, 233
pack_dms_UIMUnblockControlKey_t, 234
facilityState
pack_dms_UIMSetControlKeyProtection_t, 233
unpack_dms_UIMGetControlKeyStatus_t, 430
failureCount
FMSImageIdElement, 55
family
pack_wds_GetDefaultProfileNum_t, 329
pack_wds_SetDefaultProfileNum_t, 332
feature
uim_depersonalizationInformation, 384
uim_personalizationStatus, 390
fileID
uim_fileAttributes, 388
uim_fileInfo, 389
fileIndex
pack_uim_ReadTransparent_t, 299
pack_uim_SLQSUIMGetFileAttributes_t, 303
fileSize
uim_fileAttributes, 388
fileType
uim_fileAttributes, 388
fill_pack_ctx
common.h, 657
fill_sdu_hdr
common.h, 657
filterId
LibPackTFTIDParams, 84
fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 294
unpack_swiloc_SwiLocGetAutoStart_t, 533
fix_rate_reported
unpack_swiloc_SwiLocGetAutoStart_t, 533
fix_type
pack_swiloc_SwiLocSetAutoStart_t, 294
unpack_swiloc_SwiLocGetAutoStart_t, 533
fix_type_reported
unpack_swiloc_SwiLocGetAutoStart_t, 533
flags
sensorData_t, 352
flowLabel
LibPackTFTIDParams, 84
fms.h, 709
GetValidFwPriCombinations, 710
pack_fms_GetImagesPreference, 710
pack_fms_GetStoredImages, 711
pack_fms_SetImagesPreference, 711
unpack_fms_GetImagesPreference, 711
unpack_fms_GetStoredImages, 711
unpack_fms_SetImagesPreference, 711
Forbidden
nas_QmiNas3GppNetworkInfo, 178
ForceRev0
unpack_nas_GetCDMANetworkParameters_t, 455
format
sMSTransferRouteMTMessage, 373
ForwardMac
nas_protocolSubtypeElement, 176
fqdnAddr
wds_PCSCFFQDNAddress, 644
fqdnLen
wds_PCSCFFQDNAddress, 644
freeSlots
sms_maxStorageSizeResp, 360
freq
nas_PhyCaAggPcellInfo, 167
nas_PhyCaAggScellIndType, 169
nas_PhyCaAggScellInfo, 172
NASPhyCaAggPcellInfo, 217
NASPhyCaAggScellIndType, 218
NASPhyCaAggScellInfo, 219
freqsLen
nas_LTEInfoInterfreq, 144
nas_LTEInfoNeighboringGSM, 147
nas_LTEInfoNeighboringWCDMA, 148
fromServiceId
sms_BroadcastConfig, 356
function
pack_swiloc_SwiLocSetAutoStart_t, 294
unpack_swiloc_SwiLocGetAutoStart_t, 533
function_reported
unpack_swiloc_SwiLocGetAutoStart_t, 533
FwAutoCheck
unpack_swioma_SLQSOMADMGetSettings_t, 538
FwAutoSDM
unpack_swiavms_SLQSAvMSGetSettings_t, 530
FwAutoUpdate
pack_swiavms_SLQSAvMSSetSettings_t, 291
unpack_swiavms_SLQSAvMSGetSettings_t, 530
FwAutodownload
pack_swiavms_SLQSAvMSSetSettings_t, 291
unpack_swiavms_SLQSAvMSGetSettings_t, 530
FwAvailability
unpack_swioma_SLQSOMADMStartSession_t, 539
fwdloadsize
unpack_omaDmFotaTlv_t, 496
fwloadComplete
unpack_omaDmFotaTlv_t, 496
fwvers
unpack_dms_SLQSSwiGetFirmwareCurr_t, 424

gDlBitRate
LibPackQosClassID, 82
GPRSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 590

GPSLPM
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
GPSSel
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
GSMRSSIThreshListLen
nas_GSMRSSIThresh, 128
GSMSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 481
gUlBitRate
LibPackQosClassID, 82
GWAOPTlv
NASQmiCbkNasSystemSelPrefInd, 220
GWAcqOrderPref
NASGWAcqOrderPrefTlv, 214
GWAddressV4
unpack_wds_SLQSGetRuntimeSettings_t, 590
gcdumpString
crashInformation, 48
gcdumpStrlen
crashInformation, 48
geoSysIdx
nas_AddCDMASysInfo, 102
nas_AddSysInfo, 102
geranArfcn
nas_geranInstInfo, 126
geranBsicBcc
nas_geranInstInfo, 126
geranBsicNcc
nas_geranInstInfo, 127
geranInst
nas_UMTSInfo, 203
GeranInstInfo
nas_UMTSInfo, 203
geranRssi
nas_geranInstInfo, 127
get_version
common.h, 657
getAllCallInfo
voice_arrCallInfo, 604
GetAllCallRmtPtyName
voice_arrRemotePartyName, 606
GetCustomFeatureV2
unpack_dms_GetCustFeaturesV2_t, 402
GetValidFwPriCombinations
fms.h, 710
globalCellId
nas_LTEInfoIntrafreq, 146
glog
common.h, 658
gloglvl
common.h, 658
gnssSvId
loc_satelliteInfo, 96
gnssSvUsedList
loc_svUsedforFix, 99
gnssSvUsedList_len
loc_svUsedforFix, 99
Gpp2TimeZone
unpack_nas_SLQSGetServingSystem_t, 470
GppNetworkDSTAdjustment
unpack_nas_SLQSGetServingSystem_t, 471
GppTimeZone
unpack_nas_SLQSGetServingSystem_t, 471
GpsEnable
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
gpsTimeOfWeekMs
loc_gpsTime, 92
gpsWeek
loc_gpsTime, 92
grntDownlinkBitrate
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 648
grntUplinkBitrate
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 649
gsmAmrStat
voice_curAMRConfig, 622
GsmCellInfo
nas_lteGsmCellInfo, 141
gsmUmtsDl
pack_nas_SLQSNasSwiIndicationRegister_t, 267
gsmUmtsUl
pack_nas_SLQSNasSwiIndicationRegister_t, 267
guaranteedRate
unpack_qos_dataRate_t, 499
gwAddressV6
wds_IPV6GWAddressInfo, 643
gwV6PrefixLen
wds_IPV6GWAddressInfo, 643
gyroData
pack_loc_SLQSLOCInjectSensorData_t, 248
gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 248
gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 248

HASPI
unpack_wds_GetMobileIPProfile_t, 579
HAState
unpack_wds_GetMobileIPProfile_t, 579
HDOP
loc_precisionDilution, 94
HDRECIOThreshListLen
nas_HDRECIOThresh, 132
HDRIOThreshListLen
nas_HDRIOThresh, 133
HDRRSSIThreshListLen
nas_HDRRSSIThresh, 133
HDRSINRThresListLen
nas_HDRSINRThresh, 134
HDRSINRThreshListLen
nas_HDRSINRThreshold, 135
HDRSSInfo
unpack_nas_SLQSNasGetSigInfo_t, 481

HWVersion
nas_DeviceConfigDetail, 121
HardwareControlledMode
unpack_dms_GetPower_t, 411
has_accelTemp
pack_loc_SLQSLOCInjectSensorData_t, 248
has_acceleroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 248
has_accleroData
pack_loc_SLQSLOCInjectSensorData_t, 248
has_altitudeSrcInfo
pack_loc_SLQSLOCInjectPosition_t, 245
has_altitudeWrtEllipsoid
pack_loc_SLQSLOCInjectPosition_t, 245
has_altitudeWrtMeanSeaLevel
pack_loc_SLQSLOCInjectPosition_t, 245
has_confidence
pack_loc_SLQSLOCSetCradleMountConfig_t, 249
has_gyroData
pack_loc_SLQSLOCInjectSensorData_t, 248
has_gyroTemp
pack_loc_SLQSLOCInjectSensorData_t, 248
has_gyroTimeSrc
pack_loc_SLQSLOCInjectSensorData_t, 248
has_horConfidence
pack_loc_SLQSLOCInjectPosition_t, 245
has_horReliability
pack_loc_SLQSLOCInjectPosition_t, 245
has_horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 245
has_latitude
pack_loc_SLQSLOCInjectPosition_t, 246
has_longitude
pack_loc_SLQSLOCInjectPosition_t, 246
has_opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 248
has_positionSrc
pack_loc_SLQSLOCInjectPosition_t, 246
has_rawHorConfidence
pack_loc_SLQSLOCInjectPosition_t, 246
has_rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 246
has_timestampAge
pack_loc_SLQSLOCInjectPosition_t, 246
has_timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 246
has_vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 246
has_vertRelicability
pack_loc_SLQSLOCInjectPosition_t, 246
has_vertUnc
pack_loc_SLQSLOCInjectPosition_t, 246
hdrActiveProt
nas_HDRSysInfo, 137
hdrActiveProtValid
nas_HDRSysInfo, 137
hdrHybrid
nas_detailSvcInfo, 120
HdrPersonality
unpack_nas_SLQSGetServingSystem_t, 471
hdrPersonality
nas_HDRSysInfo, 137
NASServingSystemInfo, 222
hdrPersonalityValid
nas_HDRSysInfo, 137
hdrSSInfo, 57
ecio, 58
io, 58
rssi, 58
sinr, 58
hdrSrvStatus
nas_detailSvcInfo, 120
healthStatus
loc_satelliteInfo, 96
helper_get_resp_ctx
common.h, 657
helper_get_xid
common.h, 657
helper_set_log_func
common.h, 657
helper_set_log_lvl
common.h, 657
homeOrigVoiceSO
voice_prefVoiceSO, 630
homePageVoiceSO
voice_prefVoiceSO, 630
horConfidence
pack_loc_SLQSLOCInjectPosition_t, 246
horReliability
pack_loc_SLQSLOCInjectPosition_t, 246
horUncCircular
pack_loc_SLQSLOCInjectPosition_t, 246
hotSwap
uim_hotSwapStatus, 389
hotSwapLength
uim_hotSwapStatus, 389
hour
nas_timeInfo, 200
nas_UniversalTime, 206
hsCallStatus
nas_WCDMASysInfo, 212
hsCallStatusValid
nas_WCDMASysInfo, 212
hsInd
nas_WCDMASysInfo, 212
hsIndValid
nas_WCDMASysInfo, 212
hwType
wds_DHCPv4HWConfig, 639
wdsDhcpv4HwConfig, 649
hwVer
unpack_dms_GetHardwareRevision_t, 408

iLTEbandValue
nas_PhyCaAggPcellInfo, 167
nas_PhyCaAggScellInfo, 172
NASPhyCaAggPcellInfo, 217

NASPhyCaAggScellInfo, 219
IMCNflag
unpack_wds_SLQSGetRuntimeSettings_t, 590
IMEIString
unpack_dms_GetDeviceSerialNumbers_t, 405
IMG_MASK_CLEAR
lite-fw.h, 713
IMG_MASK_GENERIC
lite-fw.h, 714
IMG_MASK_MDM
lite-fw.h, 714
IMG_MASK_PRI
lite-fw.h, 714
IMSInfo
sMSOnIMSTlv, 372
IMSTlv
unpack_sms_SetNewSMSCallback_ind_t, 523
INT32
SwiDataTypes.h, 878
INT8
SwiDataTypes.h, 878
IOThresListLen
nas_IOThresh, 139
IPAddress
nas_DataStatusDetail, 118
IPAddressV6
ipv6AddressInfo, 58
wds_IPV6AddressInfo, 643
IPFamSupport
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
IPFamilyPreference
pack_wds_SLQSSetIPFamilyPreference_t, 342
unpack_wds_SLQSGetRuntimeSettings_t, 590
IPSECSPI
LibPackTFTIDParams, 84
IPV6AddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 590
IPV6GWAddrInfo
unpack_wds_SLQSGetRuntimeSettings_t, 590
IPV6PrefixLen
ipv6AddressInfo, 59
wds_IPV6AddressInfo, 643
IPv4
unpack_wds_SLQSGetRuntimeSettings_t, 590
IPv4Addr
wds_IPv4AdTlv, 642
IPv4AddrTlv
unpack_wds_DHCPv4ClientLease_ind_t, 574
IPv4DstAddr
unpack_qos_swiQosFilter_t, 512
IPv4SrcAddr
unpack_qos_swiQosFilter_t, 512
IPv4Tos
unpack_qos_swiQosFilter_t, 512
IPv6DstAddr
unpack_qos_swiQosFilter_t, 512
IPv6Label
unpack_qos_swiQosFilter_t, 512
IPv6SrcAddr
unpack_qos_swiQosFilter_t, 512
IPv6TrafCls
unpack_qos_swiQosFilter_t, 512
Id
unpack_qos_swiQosFilter_t, 512
id
loc_BdsSV, 88
loc_SV, 98
nas_CSGID, 114
pack_dms_UIMChangePIN_t, 231
pack_dms_UIMSetPINProtection_t, 233
pack_dms_UIMUnblockPIN_t, 234
pack_dms_UIMVerifyPIN_t, 235
unpack_qos_QosFlowInfoState_t, 504
unpack_qos_SLQSSetQosStatusCallback_ind_t, 509
id_length
DMScustSettingInfo, 51
IdleState
nas_protocolSubtypeElement, 176
image_info_t, 58
buildID, 58
buildIDLen, 58
imageType, 58
uniqueID, 58
imageID
FMSImageIdElement, 55
imageIDElement
FMSImageIDEntries, 56
imageIDEntries
FMSImageList, 57
imageIDSize
FMSImageIDEntries, 56
imageId
FMSImageElement, 54
imageList
unpack_fms_GetStoredImages_t, 435
ImageListSize
unpack_fms_GetImagesPreference_t, 434
imageListSize
pack_fms_SetImagesPreference_t, 237
imageType
FMSImageElement, 54
FMSImageIDEntries, 56
image_info_t, 58
ImageTypes
unpack_fms_SetImagesPreference_t, 436
ImageTypesSize
unpack_fms_SetImagesPreference_t, 436
imagelistSize
unpack_fms_GetStoredImages_t, 435
imei_no
unpack_dms_GetSerialNumbers_t, 412
imeiSize
unpack_dms_GetDeviceSerialNumbers_t, 405
imeiSvnSize

unpack_dms_GetDeviceSerialNumbers_t, 405
ImeiSvnString
unpack_dms_GetDeviceSerialNumbers_t, 405
imeisv_svn
unpack_dms_GetSerialNumbers_t, 412
imgType
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 425
imsRegState
nas_CommInfo, 114
imsi
unpack_dms_GetIMSI_t, 408
imsi_11_12
nas_CDMASysInfoExt, 111
imsiM1112
nas_minBasedIMSI, 158
imsiMS1
nas_minBasedIMSI, 158
imsiMS2
nas_minBasedIMSI, 158
imsiT1112
nas_trueIMSI, 201
imsiTS1
nas_trueIMSI, 201
imsiTS2
nas_trueIMSI, 201
imsiTaddrNum
nas_trueIMSI, 201
InUse
nas_QmiNas3GppNetworkInfo, 178
includes_pcs_digit
nas_QmisNasPcsDigit, 179
index
pack_wds_GetMobileIPProfile_t, 330
pack_wds_SetDefaultProfileNum_t, 332
pack_wds_SetMobileIPProfile_t, 335
unpack_qos_swiQosFilter_t, 512
unpack_qos_swiQosFlow_t, 516
unpack_wds_GetDefaultProfileNum_t, 578
index1xPri
uim_cardStatus, 382
index1xSec
uim_cardStatus, 382
indexGwPri
uim_cardStatus, 382
indexGwSec
uim_cardStatus, 382
Info
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 485
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 486
InfoInterfreq
nas_LTEInfoInterfreq, 144
insNmrCellInfo
nas_GERANInfo, 126
instancesSize
unpack_nas_GetRFInfo_t, 459
interval
pack_pds_SetPDSDefaults_t, 276
pack_wds_SLQSSetWdsEventCallback_t, 343
Io
unpack_nas_SLQSGetSignalStrength_t, 472
io
hdrSSInfo, 58
nas_SLQSSignalStrengthsInformation, 192
ioDelta
nas_SLQSSignalStrengthsIndReq, 192
ipAddress
pack_wds_SetDefaultProfile_t, 332
ipFamily
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 592
ipVersion
LibPackTFTIDParams, 84
ipaddr
unpack_wds_GetDefaultProfile_t, 577
ipaddrv6
unpack_wds_GetDefaultProfile_t, 577
ipv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
ipv4GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
ipv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
ipv6AddressInfo, 58
IPAddressV6, 58
IPV6PrefixLen, 59
ipv6GWAddress
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
is856SysId
nas_HDRSysInfo, 137
is856SysIdValid
nas_HDRSysInfo, 137
is_DataRate_Available
unpack_qos_swiQosFlow_t, 516
is_EspSpi_Available
unpack_qos_swiQosFilter_t, 512
is_IPv4DstAddr_Available
unpack_qos_swiQosFilter_t, 512
is_IPv4SrcAddr_Available
unpack_qos_swiQosFilter_t, 512
is_IPv4Tos_Available
unpack_qos_swiQosFilter_t, 512
is_IPv6DstAddr_Available
unpack_qos_swiQosFilter_t, 512
is_IPv6Label_Available
unpack_qos_swiQosFilter_t, 512
is_IPv6SrcAddr_Available
unpack_qos_swiQosFilter_t, 512
is_IPv6TrafCls_Available
unpack_qos_swiQosFilter_t, 512
is_Id_Available

unpack_qos_swiQosFilter_t, 512
is_Jitter_Available
unpack_qos_swiQosFlow_t, 516
is_Latency_Available
unpack_qos_swiQosFlow_t, 516
is_LteBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 421
is_LteQci_Available
unpack_qos_swiQosFlow_t, 516
is_MaxAllowedPktSz_Available
unpack_qos_swiQosFlow_t, 516
is_MinPolicedPktSz_Available
unpack_qos_swiQosFlow_t, 516
is_NxtHdrProto_Available
unpack_qos_swiQosFilter_t, 512
is_PktErrRate_Available
unpack_qos_swiQosFlow_t, 516
is_Precedence_Available
unpack_qos_swiQosFilter_t, 512
is_ProfileId3GPP2_Available
unpack_qos_swiQosFlow_t, 516
is_RxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 503
is_TCPDstPort_Available
unpack_qos_swiQosFilter_t, 512
is_TCPSrcPort_Available
unpack_qos_swiQosFilter_t, 512
is_TdsBandCapability_Available
unpack_dms_SLQSGetBandCapability_t, 421
is_TokenBucket_Available
unpack_qos_swiQosFlow_t, 516
is_TrafficClass_Available
unpack_qos_swiQosFlow_t, 516
is_TranDstPort_Available
unpack_qos_swiQosFilter_t, 513
is_TranSrcPort_Available
unpack_qos_swiQosFilter_t, 513
is_TxQFlowGranted_Available
unpack_qos_QosFlowInfo_t, 503
is_UDPDstPort_Available
unpack_qos_swiQosFilter_t, 513
is_UDPSrcPort_Available
unpack_qos_swiQosFilter_t, 513
is_val_3GPP2Pri_Available
unpack_qos_swiQosFlow_t, 516
is_val_3GPPImCn_Available
unpack_qos_swiQosFlow_t, 516
is_val_3GPPResResidualBER_Available
unpack_qos_swiQosFlow_t, 516
is_val_3GPPSigInd_Available
unpack_qos_swiQosFlow_t, 516
is_val_3GPPTraHdlPri_Available
unpack_qos_swiQosFlow_t, 516
isEmpty
voice_getAllCallInformation, 625
isInTraffic
nas_txInfo, 202
isNewFlow
unpack_qos_QosFlowInfoState_t, 504
isPrefDataPath
nas_GSMSrvStatusInfo, 129
nas_SrvStatusInfo, 194
isRadioTuned
nas_rxInfo, 185
nas_RxSigInfo, 186
isSysForbidden
nas_detailSvcInfo, 120
nas_sysInfoCommon, 196
isSysForbiddenValid
nas_sysInfoCommon, 196
isSysPrlMatch
nas_CDMASysInfo, 110
nas_HDRSysInfo, 137
isSysPrlMatchValid
nas_CDMASysInfo, 110
nas_HDRSysInfo, 137
IsVoiceEnabled
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402

Jitter
unpack_qos_swiQosFlow_t, 516

KeyExchange
nas_protocolSubtypeElement, 176

LITEQMI_QMI_CBK_PARAM_NOCHANGE
sms.h, 842
LITEQMI_QMI_CBK_PARAM_RESET
sms.h, 842
LITEQMI_QMI_CBK_PARAM_SET
sms.h, 842
LBPTlv
NASQmiCbkNasSystemSelPrefInd, 221
LOCEVENTMASKGNSSSVINFO
loc.h, 723
LOCEVENTMASKNMEA
loc.h, 724
LOCEVENTMASKWIFIREQ
loc.h, 724
LPCSTR
SwiDataTypes.h, 878
LTEAttachProfile
unpack_wds_SLQSGet3GPPConfigItem_t, 585
LTEAttachProfileList
unpack_wds_SLQSGet3GPPConfigItem_t, 585
LTEAttachProfileListLen
pack_wds_SLQSSet3GPPConfigItem_t, 341
unpack_wds_SLQSGet3GPPConfigItem_t, 585
LTEBandPref
NASLTEBandPreferenceTlv, 214
LTECphyCaInfo
unpack_nas_SlqsGetLTECphyCAInfo_t, 466
LTERSRPThreshListLen
nas_LTERSRPThresh, 149
LTERSRQThreshListLen
nas_LTERSRQThresh, 150

LTERSSIThreshListLen
nas_LTERSSIThresh, 150
LTESNRThresListLen
nas_LTESNRThresh, 153
LTESNRThreshListLen
nas_LTESNRThreshold, 154
LTESSInfo
unpack_nas_SLQSNasGetSigInfo_t, 481
LTEWCDMACellInfo
nas_LTEInfoNeighboringWCDMA, 148
Lac
unpack_nas_SLQSGetServingSystem_t, 471
lac
nas_GERANInfo, 126
nas_GSMSysInfo, 132
nas_LTESysInfo, 156
nas_UMTSInfo, 203
nas_WCDMASysInfo, 212
lac1
nas_OperatorPLMNData, 166
lac2
nas_OperatorPLMNData, 166
lacValid
nas_GSMSysInfo, 132
nas_LTESysInfo, 156
nas_WCDMASysInfo, 212
language
sms_CDMABroadcastConfig, 356
lastCallDataBearerTech
unpack_wds_SLQSGetDUNCallInfo_t, 588
lastCallDataBearerTechnology
unpack_wds_SLQSGetDataBearerTechnology_t, 587
lastCallRXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 588
lastCallTXOKBytesCnt
unpack_wds_SLQSGetDUNCallInfo_t, 588
LastErrCode
nas_DataStatusDetail, 118
Latency
unpack_qos_swiQosFlow_t, 516
latitude
pack_loc_SLQSLOCInjectPosition_t, 246
leapSeconds
nas_qaQmi3Gpp2TimeZone, 177
leaseState
wds_DHCPLeaseStateTlv, 637
len
loc_BdsSVInfo, 88
loc_SVInfo, 98
unpack_nas_GetSignalStrengths_t, 461
length
sMSCAddress, 369
sMSEtwsMessage, 370
sMSTransferRouteMTMessage, 373
uim_readTransparentInfo, 391
LibPackGPRSRequestedQoS, 59
delayClass, 59
meanThroughputClass, 59
peakThroughputClass, 59
precedenceClass, 59
reliabilityClass, 59
LibPackQosClassID, 82
gDlBitRate, 82
gUlBitRate, 82
maxDlBitRate, 82
maxUlBitRate, 82
QCI, 82
LibPackTFTIDParams, 82
destPortRangeEnd, 84
destPortRangeStart, 84
eValid, 84
filterId, 84
flowLabel, 84
IPSECSPI, 84
ipVersion, 84
nextHeader, 84
pSourceIP, 84
sourceIPMask, 84
srcPortRangeEnd, 84
srcPortRangeStart, 84
tosMask, 84
LibPackUMTSQoS, 84
deliveryErrSDU, 86
grntDownlinkBitrate, 86
grntUplinkBitrate, 86
maxDownlinkBitrate, 86
maxSDUSize, 86
maxUplinkBitrate, 86
qosDeliveryOrder, 86
resBerRatio, 86
sduErrorRatio, 87
trafficClass, 87
trafficPriority, 87
transferDelay, 87
LibPackUMTSReqQoSSigInd, 87
SigInd, 87
UMTSReqQoS, 87
LibPackprofile_3GPP, 70
pAPNClass, 75
pAPNDisabledFlag, 75
pAPNName, 75
pAPNnameSize, 75
pAddrAllocPref, 75
pAuthenticationPref, 75
pGPRSMinimumQoS, 75
pGPRSRequestedQos, 75
pIPv4AddrPref, 75
pIPv6AddPref, 75
pImCnFlag, 75
pPDNInactivTimeout, 76
pPDPtype, 76
pPassword, 75
pPasswordSize, 75
pPcscfAddrUsingDhcp, 75
pPcscfAddrUsingPCO, 76

pPdpAccessConFlag, 76
pPdpContext, 76
pPdpDataCompType, 76
pPdpHdrCompType, 76
pPriDNSIPv4AddPref, 76
pPriDNSIPv6addpref, 76
pPrimaryID, 76
pProfilename, 76
pProfilenameSize, 76
pQosClassID, 76
pSecDNSIPv4AddPref, 76
pSecDNSIPv6addpref, 76
pSecondaryFlag, 76
pTFTID1Params, 76
pTFTID2Params, 76
pUMTSMinQoS, 76
pUMTSMinQosSigInd, 76
pUMTSReqQoS, 76
pUMTSReqQoSSigInd, 76
pUsername, 76
pUsernameSize, 76
LibPackprofile_3GPP2, 76
pAPNClass3GPP2, 80
pAPNEnabled3GPP2, 80
pAllowLinger, 80
pApnString, 81
pApnStringSize, 81
pAppPriority, 81
pAppType, 81
pAuthPassword, 81
pAuthPassword_tSize, 81
pAuthProtocol, 81
pAuthRetryCount, 81
pAuthTimeout, 81
pDataMode, 81
pDataRate, 81
pIpcpAckTimeout, 81
pIpcpCreqRetryCount, 81
pIsPcscfAddressNedded, 81
pLcpAckTimeout, 81
pLcpCreqRetryCount, 81
pNegoDnsSrvrPref, 81
pPDNInactivTimeout3GPP2, 81
pPdnType, 81
pPppSessCloseTimer1x, 81
pPppSessCloseTimerDO, 81
pPriV6DnsAddress, 81
pPrimaryV4DnsAddress, 81
pRATType, 81
pSecV6DnsAddress, 81
pSecondaryV4DnsAddress, 81
pUserId, 81
pUserIdSize, 81
libSDP_BuildImagesPreferenceRequest
lite-fw.h, 714
libSDP_CalculateImageMask
lite-fw.h, 714
libSDP_CheckValidFirmwareInfo
lite-fw.h, 714
libSDP_DownloadFW
lite-fw.h, 714
libSDP_ExtractFirmwareParametersByPath
lite-fw.h, 714
libSDP_FirmwareInfo
lite-fw.h, 714
libSDP_GetModelFamily
lite-fw.h, 714
libSDP_GetVersion
lite-fw.h, 714
libSDP_getFileType
lite-fw.h, 714
LibpackProfile3GPP, 59
pAPNClass, 64
pAPNDisabledFlag, 64
pAPNName, 64
pAPNnameSize, 64
pAddrAllocPref, 64
pAuthenticationPref, 64
pGPRSMinimumQoS, 64
pGPRSRequestedQos, 64
pIPv4AddrPref, 64
pIPv6AddPref, 64
pImCnFlag, 64
pPDNInactivTimeout, 64
pPDPtype, 65
pPassword, 64
pPasswordSize, 64
pPcscfAddrUsingDhcp, 64
pPcscfAddrUsingPCO, 64
pPdpAccessConFlag, 64
pPdpContext, 64
pPdpDataCompType, 64
pPdpHdrCompType, 65
pPriDNSIPv4AddPref, 65
pPriDNSIPv6addpref, 65
pPrimaryID, 65
pProfilename, 65
pProfilenameSize, 65
pQosClassID, 65
pSecDNSIPv4AddPref, 65
pSecDNSIPv6addpref, 65
pSecondaryFlag, 65
pTFTID1Params, 65
pTFTID2Params, 65
pUMTSMinQoS, 65
pUMTSMinQosSigInd, 65
pUMTSReqQoS, 65
pUMTSReqQoSSigInd, 65
pUsername, 65
pUsernameSize, 65
LibpackProfile3GPP2, 65
pAPNClass3GPP2, 69
pAPNEnabled3GPP2, 69
pAllowLinger, 69
pApnString, 69
pApnStringSize, 69

pAppPriority, 70
pAppType, 70
pAuthPassword, 70
pAuthPasswordSize, 70
pAuthProtocol, 70
pAuthRetryCount, 70
pAuthTimeout, 70
pDataMode, 70
pDataRate, 70
pIpcpAckTimeout, 70
pIpcpCreqRetryCount, 70
pIsPcscfAddressNedded, 70
pLcpAckTimeout, 70
pLcpCreqRetryCount, 70
pNegoDnsSrvrPref, 70
pPDNInactivTimeout3GPP2, 70
pPdnType, 70
pPppSessCloseTimer1x, 70
pPppSessCloseTimerDO, 70
pPriV6DnsAddress, 70
pPrimaryV4DnsAddress, 70
pRATType, 70
pSecV6DnsAddress, 70
pSecondaryV4DnsAddress, 70
pUserId, 70
pUserIdSize, 70
libsdp_SetReadBlockSize
lite-fw.h, 714
libsdp_set_log_func
lite-fw.h, 714
libsdplogger
lite-fw.h, 714
lineValue
pack_voice_SLQSVoiceALSSelectLine_t, 311
linkage
altSrcInfo_t, 41
list_type
DMScustSettingList, 51
DMSgetCustomInput, 52
pack_dms_GetCustFeaturesV2_t, 224
listEntries
FMSPrefImageList, 57
listSize
FMSImageList, 57
FMSPrefImageList, 57
lite-fw.h
eFW_TYPE_CWE, 715
eFW_TYPE_CWE_NVU, 715
eFW_TYPE_INVALID, 715
eFW_TYPE_MBN, 715
eFW_TYPE_MBN_GOBI, 715
eFW_TYPE_NVU, 715
eFW_TYPE_SPK, 715
eModel_9X15, 716
eModel_9X30, 716
eModel_Unknown, 716
eSDP_FWDWL_ERR_END, 716
eSDP_FWDWL_ERR_FAIL, 716
eSDP_FWDWL_ERR_FW_UPGRADE, 715
eSDP_FWDWL_ERR_FW_VERSION_FAIL, 716
eSDP_FWDWL_ERR_GENERAL, 715
eSDP_FWDWL_ERR_INVALID_DEV, 716
eSDP_FWDWL_ERR_INVALID_PATH, 716
eSDP_FWDWL_ERR_PATH_NOT_SPECIFIED, 715
eSDP_FWDWL_ERR_PATH_TOO_LONG, 715
eSDP_FWDWL_ERR_PRI_FAIL, 716
eSDP_FWDWL_ERR_SDK, 715
eSDP_FWDWL_ERR_SET_CBK, 715
eSDP_FWDWL_ERR_TIMEOUT, 716
eSDP_FWDWL_SUCCESS, 715
lite-fw.h, 712
IMG_MASK_CLEAR, 713
IMG_MASK_GENERIC, 714
IMG_MASK_MDM, 714
IMG_MASK_PRI, 714
libSDP_BuildImagesPreferenceRequest, 714
libSDP_CalculateImageMask, 714
libSDP_CheckValidFirmwareInfo, 714
libSDP_DownloadFW, 714
libSDP_ExtractFirmwareParametersByPath, 714
libSDP_FirmwareInfo, 714
libSDP_GetModelFamily, 714
libSDP_GetVersion, 714
libSDP_getFileType, 714
libsdp_SetReadBlockSize, 714
libsdp_set_log_func, 714
libsdplogger, 714
litefw_BuildImagesPreferenceRequest, 716
litefw_CalculateImageMask, 716
litefw_CheckValidFirmwareInfo, 717
litefw_DownloadFW, 717
litefw_ExtractFirmwareParametersByPath, 717
litefw_FirmwareInfo, 714
litefw_Fw_Type, 715
litefw_GetModelFamily, 718
litefw_GetVersion, 719
litefw_Models, 716
litefw_SetReadBlockSize, 719
litefw_fwdwl_error_codes, 715
litefw_getFileType, 718
litefw_set_log_func, 719
litefwlogger, 715
litefw_BuildImagesPreferenceRequest
lite-fw.h, 716
litefw_CalculateImageMask
lite-fw.h, 716
litefw_CheckValidFirmwareInfo
lite-fw.h, 717
litefw_DownloadFW
lite-fw.h, 717
litefw_ExtractFirmwareParametersByPath
lite-fw.h, 717
litefw_FirmwareInfo
lite-fw.h, 714
litefw_Fw_Type

lite-fw.h, 715
litefw_GetModelFamily
lite-fw.h, 718
litefw_GetVersion
lite-fw.h, 719
litefw_Models
lite-fw.h, 716
litefw_SetReadBlockSize
lite-fw.h, 719
litefw_fwdwl_error_codes
lite-fw.h, 715
litefw_getFileType
lite-fw.h, 718
litefw_set_log_func
lite-fw.h, 719
litefwlogger
lite-fw.h, 715
liteqmi_GetVersion
common.h, 657
liteqmi_log
common.h, 657
loc.h
eQMI_LOC_SESS_STATUS_FAILURE, 725
eQMI_LOC_SESS_STATUS_IN_PROGRESS, 724
eQMI_LOC_SESS_STATUS_SUCCESS, 724
eQMI_LOC_SESS_STATUS_TIMEOUT, 725
loc.h, 719
LOCEVENTMASKNMEA, 724
LOCEVENTMASKWIFIREQ, 724
MAX_TEMP_DATA_LEN, 724
pack_loc_DeleteAssistData, 725
pack_loc_EventRegister, 725
pack_loc_SLQSLOCGetBestAvailPos, 726
pack_loc_SLQSLOCInjectPosition, 726
pack_loc_SLQSLOCInjectSensorData, 727
pack_loc_SLQSLOCInjectUTCTime, 727
pack_loc_SLQSLOCSetCradleMountConfig, 727
pack_loc_SetExtPowerState, 725
pack_loc_SetOperationMode, 726
pack_loc_Start, 728
pack_loc_Stop, 728
unpack_loc_BestAvailPos_Ind, 728
unpack_loc_DeleteAssistData, 729
unpack_loc_DeleteAssistData_Ind, 729
unpack_loc_EngineState_Ind, 729
unpack_loc_EventRegister, 730
unpack_loc_GnssSvInfo_Ind, 730
unpack_loc_PositionRpt_Ind, 730
unpack_loc_SLQSLOCGetBestAvailPos, 732
unpack_loc_SLQSLOCInjectPosition, 732
unpack_loc_SLQSLOCInjectSensorData, 733
unpack_loc_SLQSLOCInjectUTCTime, 733
unpack_loc_SLQSLOCSetCradleMountConfig, 733
unpack_loc_SetExtPowerConfig_Ind, 731
unpack_loc_SetExtPowerState, 731
unpack_loc_SetOperationMode, 731
unpack_loc_SetOperationMode_Ind, 732
unpack_loc_Start, 733
unpack_loc_Stop, 734
loc_BdsSV, 87
id, 88
mask, 88
loc_BdsSVInfo, 88
len, 88
pSV, 88
loc_CellDb, 88
mask, 89
loc_ClkInfo, 89
mask, 90
loc_GnssData, 90
mask, 91
loc_LocApplicationInfo, 92
appNameLength, 93
appProviderLength, 93
appVersionLength, 93
appVersionValid, 93
pAppName, 93
pAppProvider, 93
pAppVersion, 93
loc_SV, 97
id, 98
mask, 98
system, 98
loc_SVInfo, 98
len, 98
pSV, 99
loc_gpsTime, 92
gpsTimeOfWeekMs, 92
gpsWeek, 92
loc_precisionDilution, 93
HDOP, 94
PDOP, 94
VDOP, 94
loc_satelliteInfo, 94
azimuth, 96
elevation, 96
gnssSvId, 96
healthStatus, 96
snr, 96
svInfoMask, 96
svListLen, 96
svStatus, 96
system, 96
validMask, 96
loc_sensorDataUsage, 96
aidingIndicatorMask, 97
usageMask, 97
loc_svUsedforFix, 99
gnssSvUsedList, 99
gnssSvUsedList_len, 99
localTimeOffset
nas_qaQmi3Gpp2TimeZone, 177
logString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 425

logger
common.h, 655
longName
nas_PLMNNetworkNameData, 175
unpack_nas_SLQSGetPLMNName_t, 468
longNameCI
unpack_nas_SLQSGetPLMNName_t, 468
longNameEn
unpack_nas_SLQSGetPLMNName_t, 468
longNameLen
nas_PLMNNetworkNameData, 175
unpack_nas_SLQSGetPLMNName_t, 468
longNameSB
unpack_nas_SLQSGetPLMNName_t, 468
longNameSpareBits
nas_PLMNNetworkNameData, 175
longitude
pack_loc_SLQSLOCInjectPosition_t, 246
loopbackMode
pack_wds_SLQSSSetLoopback_t, 344
loopbackMultiplier
pack_wds_SLQSSSetLoopback_t, 344
LteBandCapability
unpack_dms_SLQSGetBandCapability_t, 421
lteEmmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 267
lteEmmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 267
lteEsmDl
pack_nas_SLQSNasSwiIndicationRegister_t, 267
lteEsmUl
pack_nas_SLQSNasSwiIndicationRegister_t, 267
LteGsmCellInfo
nas_LTEInfoNeighboringGSM, 147
LteQci
unpack_qos_swiQosFlow_t, 517
lteRsrpDelta
nas_SLQSSignalStrengthsIndReq, 192
lteRsrpinfo
nas_SLQSSignalStrengthsInformation, 192
lteSSInfo, 99
rsrp, 100
rsrq, 100
rssi, 100
snr, 100
lteSnrDelta
nas_SLQSSignalStrengthsIndReq, 192
lteSnrinfo
nas_SLQSSignalStrengthsInformation, 193
ltersrp
unpack_nas_SLQSGetSignalStrength_t, 472
ltesnr
unpack_nas_SLQSGetSignalStrength_t, 472

m_FwBuildId
CarrierImage_t, 45
m_FwImageId
CarrierImage_t, 45
m_PriBuildId
CarrierImage_t, 45
m_PriImageId
CarrierImage_t, 45
m_nCarrierId
CarrierImage_t, 45
m_nFolderId
CarrierImage_t, 45
m_nStorage
CarrierImage_t, 45
MACIndex
nas_NetworkStatEVDO, 163
MAX_BUILD_ID_LEN
dms.h, 665
MAX_ICCID_LENGTH
uim.h, 892
MAX_MSE_TWS_MSG
sms.h, 840
MAX_NO_OF_SLOTS
uim.h, 892
MAX_SENSOR_DATA_LEN
loc.h, 724
MAX_SLOTS_STATUS
uim.h, 892
MAX_SMS_LIST_SIZE
sms.h, 840
MAX_TEMP_DATA_LEN
loc.h, 724
MAXVOICEUSSDLENGTH
voice.h, 911
MCC
nas_CDMASysInfo, 110
nas_CDMASysInfoExt, 111
nas_currentPLMN, 115
nas_GSMSysInfo, 132
nas_LTESysInfo, 156
nas_QmiNas3GppNetworkInfo, 178
nas_QmiNas3GppNetworkRAT, 179
nas_QmisNasPcsDigit, 179
nas_WCDMASysInfo, 212
unpack_nas_GetServingNetwork_t, 460
MDMCallDuration
connectionStatus, 46
MDMConnStatus
connectionStatus, 46
MEID_MAX_SIZE
dms.h, 665
MEIDString
unpack_dms_GetDeviceSerialNumbers_t, 405
MIN
unpack_dms_GetVoiceNumber_t, 412
MINREQBKLEN
common.h, 655
MMTlv
unpack_sms_SetNewSMSCallback_ind_t, 523
MNC
nas_CDMASysInfo, 110
nas_currentPLMN, 115
nas_GSMSysInfo, 132

nas_LTESysInfo, 156
nas_QmiNas3GppNetworkInfo, 178
nas_QmiNas3GppNetworkRAT, 179
nas_QmisNasPcsDigit, 179
nas_WCDMASysInfo, 212
unpack_nas_GetServingNetwork_t, 460
MPTlv
NASQmiCbkNasSystemSelPrefInd, 221
MSGID_AND_LEN
common.h, 655
MSGID_DONT_CARE
common.h, 655
MTMessageInfo
newMTMessageTlv, 223
manString
pack_dms_SLQSSwiSetHostDevInfo_t, 230
unpack_dms_SLQSSwiGetHostDevInfo_t, 426
manufacturer
unpack_dms_GetManufacturer_t, 408
Mask
pack_wds_SLQSGetDUNCallInfo_t, 337
mask
loc_BdsSV, 88
loc_CellDb, 89
loc_ClkInfo, 90
loc_GnssData, 91
loc_SV, 98
unpack_qos_IPv6TrafCls_t, 501
unpack_qos_Tos_t, 518
max
_PackSwiAVMSSettingsPeriodsInfo, 30
max_channel_rx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 587
max_channel_tx_rate
unpack_wds_SLQSGetCurrentChannelRate_t, 587
max_dist
pack_swiloc_SwiLocSetAutoStart_t, 294
unpack_swiloc_SwiLocGetAutoStart_t, 533
max_dist_reported
unpack_swiloc_SwiLocGetAutoStart_t, 533
max_time
pack_swiloc_SwiLocSetAutoStart_t, 294
unpack_swiloc_SwiLocGetAutoStart_t, 534
max_time_reported
unpack_swiloc_SwiLocGetAutoStart_t, 534
MaxAllowedPktSz
unpack_qos_swiQosFlow_t, 517
MaxChanRxRate
dunchannelRate, 53
MaxChanTxRate
dunchannelRate, 53
maxChannelRXRate
unpack_wds_GetConnectionRate_t, 575
maxChannelTXRate
unpack_wds_GetConnectionRate_t, 575
maxDlBitRate
LibPackQosClassID, 82
maxDownlinkBitrate
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 649
maxImages
FMSImageIDEntries, 56
MaxRXChannelRate
unpack_dms_GetDeviceCap_t, 403
maxRxChannelRate
unpack_dms_GetDeviceCapabilities_t, 404
maxSDUSize
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 649
maxStorageSize
sms_maxStorageSizeResp, 360
MaxTXChannelRate
unpack_dms_GetDeviceCap_t, 403
maxTxChannelRate
unpack_dms_GetDeviceCapabilities_t, 404
maxUlBitRate
LibPackQosClassID, 82
maxUplinkBitrate
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 649
mcc
nas_CSGID, 115
nas_MNRInfo, 159
nas_netSelectionPref, 160
nas_OperatorPLMNData, 166
pack_nas_SLQSGetPLMNName_t, 257
unpack_nas_GetHomeNetwork_t, 457
mccM
nas_minBasedIMSI, 158
mccT
nas_trueIMSI, 201
mdmCallDurationActive
unpack_wds_SLQSGetDUNCallInfo_t, 588
meanThroughputClass
LibPackGPRSRequestedQoS, 59
wds_GPRSQoS, 642
meid
unpack_dms_GetSerialNumbers_t, 412
meidSize
unpack_dms_GetDeviceSerialNumbers_t, 405
meidString
unpack_dms_SLQSSwiGetSerialNoExt_t, 427
message
unpack_sms_SLQSGetSMS_t, 525
message_type
NASOTAMessageTlv, 216
messageClass
sms_routeEntry, 364
messageFailureCode
unpack_sms_SendSMS_t, 522
messageFormat
pack_sms_SaveSMS_t, 281
pack_sms_SendSMS_t, 281
sms_sendAsyncsmsParams, 366

unpack_sms_SLQSGetSMS_t, 525
messageID
unpack_sms_SendSMS_t, 522
messageIndex
pack_sms_SLQSGetSMS_t, 284
pack_sms_SLQSModifySMSStatus_t, 286
qmiSmsMessageList, 350
sMSMTMessage, 372
messageList
unpack_sms_SLQSGetSMSList_t, 526
messageListSize
unpack_sms_SLQSGetSMSList_t, 526
messageMode
sMSMessageMode, 371
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 529
MessageModeInfo
messageModeTlv, 100
messageModeTlv, 100
MessageModeInfo, 100
TlvPresent, 100
messageSize
pack_sms_SaveSMS_t, 281
pack_sms_SendSMS_t, 281
sms_sendAsyncsmsParams, 366
unpack_sms_SLQSGetSMS_t, 525
messageTag
pack_sms_SLQSModifySMSStatus_t, 286
qmiSmsMessageList, 350
unpack_sms_SLQSGetSMS_t, 525
messageType
sms_routeEntry, 364
min
_PackSwiAVMSSettingsPeriodsInfo, 30
MinPolicedPktSz
unpack_qos_swiQosFlow_t, 517
minSize
unpack_dms_GetVoiceNumber_t, 412
minute
nas_timeInfo, 200
nas_UniversalTime, 206
mipMode
unpack_wds_GetMobileIP_t, 579
mipStatus
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
mipstatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
mnc
nas_CSGID, 115
nas_MNRInfo, 159
nas_netSelectionPref, 160
nas_OperatorPLMNData, 166
pack_nas_SLQSGetPLMNName_t, 257
unpack_nas_GetHomeNetwork_t, 457
mncPcsDigits
nas_CSGID, 115
mobileCountryCode
sMSEtwsPlmn, 371
mobileIP
pack_wds_SLQSSetWdsEventCallback_t, 343
mobileNetworkCode
sMSEtwsPlmn, 371
mode
pack_dms_SetEventReport_t, 227
pack_dms_SetPower_t, 227
pack_loc_SetOperationMode_t, 241
pack_sms_SLQSGetSmsBroadcastConfig_t, 284
pack_sms_SLQSSetSmsBroadcastActivation_t, 287
pack_sms_SLQSSetSmsBroadcastConfig_t, 288
pack_wds_SetMobileIP_t, 333
uim_refreshevent, 393
voice_callInfo, 615
ModePref
NASModePreferenceTlv, 215
modelString
pack_dms_SLQSSwiSetHostDevInfo_t, 230
unpack_dms_SLQSSwiGetHostDevInfo_t, 426
modelid
unpack_dms_GetModelID_t, 409
modelid_str
unpack_dms_GetFirmwareInfo_t, 406
modemMode
nas_CommInfo, 114
modemindex
pack_fms_SetImagesPreference_t, 237
month
nas_timeInfo, 200
nas_UniversalTime, 206
msgCount
sms_messageWaitingInfoContent, 361
msgProtocol
sms_msgProtocolResp, 361
msgType
sms_messageWaitingInfoContent, 361
msgWaitInfo
sms_getMsgWaitingInfo, 358
msgid
pack_qmi_t, 277
unpack_qmi_t, 499
msgtype
common.h, 656
Mtu
unpack_wds_SLQSGetRuntimeSettings_t, 590
MultDisc
nas_protocolSubtypeElement, 176
multiplier
unpack_qos_pktErrRate_t, 501

NAI
unpack_wds_GetMobileIPProfile_t, 579
NAS_NAM_NAME_LENGTH
nas.h, 742
NAS_PLMN_LENGTH
nas.h, 742

NASBandPreferenceTlv, 213
band_pref, 213
TlvPresent, 213
NASEmergencyModeTlv, 213
EmerMode, 214
TlvPresent, 214
NASGWAcqOrderPrefTlv, 214
GWAcqOrderPref, 214
TlvPresent, 214
NASLTEBandPreferenceTlv, 214
LTEBandPref, 214
TlvPresent, 214
NASLteNasReleaseInfoTlv, 215
nas_major, 215
nas_minor, 215
nas_release, 215
TlvPresent, 215
NASModePreferenceTlv, 215
ModePref, 215
TlvPresent, 215
NASNetSelPreferenceTlv, 215
NetSelPref, 215
TlvPresent, 215
NASOTAMessageTlv, 215
data_buf, 216
data_len, 216
message_type, 216
TlvPresent, 216
NASPRLPreferenceTlv, 220
PRLPref, 220
TlvPresent, 220
NASPhyCaAggPcellInfo, 216
dl_bw_value, 217
freq, 217
iLTEbandValue, 217
pci, 217
TlvPresent, 217
NASPhyCaAggScellDlBw, 217
dl_bw_value, 217
TlvPresent, 217
NASPhyCaAggScellIndType, 218
freq, 218
pci, 218
scell_state, 218
TlvPresent, 218
NASPhyCaAggScellIndex, 217
scell_idx, 218
TlvPresent, 218
NASPhyCaAggScellInfo, 219
dl_bw_value, 219
freq, 219
iLTEbandValue, 219
pci, 219
scell_state, 219
TlvPresent, 219
NASQmiCbkNasSwiOTAMessageInd, 220
nasRelInfoTlv, 220
otaMsgTlv, 220
timeTlv, 220
NASQmiCbkNasSystemSelPrefInd, 220
BPTlv, 220
EMTlv, 220
GWAOPTlv, 220
LBPTlv, 221
MPTlv, 221
NSPTlv, 221
PRLPTlv, 221
RPTlv, 221
SDPTlv, 221
NASRoamPreferenceTlv, 221
RoamPref, 221
TlvPresent, 221
NASServDomainPrefTlv, 221
SrvDomainPref, 221
TlvPresent, 221
NASServingSystemInfo, 221
csAttachState, 222
hdrPersonality, 222
psAttachState, 223
radioInterfaceList, 223
radioInterfaceNo, 223
registrationState, 223
selectedNetwork, 223
NASTimeInfoTlv, 223
time, 223
TlvPresent, 223
NSPTlv
NASQmiCbkNasSystemSelPrefInd, 221
NWQoSStatus
unpack_qos_SLQSQosGetNetworkStatus_t, 504
naiSize
unpack_wds_GetMobileIPProfile_t, 579
namID
pack_nas_SLQSNasGet3GPP2Subscription_t, 263
voice_airTimer, 598
voice_prefVoiceSO, 630
voice_roamTimer, 633
namName
nas_namName, 159
namNameLen
nas_namName, 159
Name
unpack_nas_GetServingNetwork_t, 460
name
unpack_nas_GetHomeNetwork_t, 457
unpack_wds_GetDefaultProfile_t, 577
nameLen
voice_remotePartyName, 631
namePI
voice_remotePartyName, 631
nameSize
unpack_nas_GetHomeNetwork3GPP2_t, 456
unpack_nas_GetServingNetwork_t, 460
nameString
pack_dms_SLQSSwiSetOSInfo_t, 230

unpack_dms_SLQSSwiGetOSInfo_t, 427
namelength
unpack_omaDmFotaTlv_t, 496
namesize
unpack_wds_GetDefaultProfile_t, 577
nas.h
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_100, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_15, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_25, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_50, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_6, 742
eLITEQMI_NAS_LTE_CPHY_CA_BW_NRB_75, 742
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED, 743
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED, 743
eLITEQMI_NAS_LTE_CPHY_SCELL_STATE_DECONFIGURED, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_100, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_15, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_25, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_50, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_6, 743
eNAS_LTE_CPHY_CA_BW_NRB_LITE_75, 743
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_ACTIVATED_LITE, 743
eNAS_LTE_CPHY_SCELL_STATE_CONFIGURED_DEACTIVATED_LITE, 743
eNAS_LTE_CPHY_SCELL_STATE_DECONFIGURED_LITE, 743
nas.h, 734
NAS_PLMN_LENGTH, 742
pack_nas_GetACCOLC, 743
pack_nas_GetANAAAAuthenticationStatus, 743
pack_nas_GetCDMANetworkParameters, 744
pack_nas_GetHomeNetwork, 744
pack_nas_GetHomeNetwork3GPP2, 744
pack_nas_GetNetworkPreference, 745
pack_nas_GetRFInfo, 745
pack_nas_GetServingNetwork, 745
pack_nas_GetServingNetworkCapabilities, 745
pack_nas_GetSignalStrengths, 746
pack_nas_InitiateDomainAttach, 746
pack_nas_PerformNetworkScan, 746
pack_nas_SLQSConfigSigInfo, 748
pack_nas_SLQSGetErrorRate, 749
pack_nas_SLQSGetNetworkTime, 749
pack_nas_SLQSGetOperatorNameData, 749
pack_nas_SLQSGetPLMNName, 750
pack_nas_SLQSGetServingSystem, 750
pack_nas_SLQSGetSignalStrength, 750
pack_nas_SLQSGetSysInfo, 751
pack_nas_SLQSGetSysSelectionPref, 751
pack_nas_SLQSInitiateNetworkRegistration, 751
pack_nas_SLQSNASSwiGetChannelLock, 754
pack_nas_SLQSNASSwiSetChannelLock, 755
pack_nas_SLQSNasConfigSigInfo2, 752
pack_nas_SLQSNasGet3GPP2Subscription, 752
pack_nas_SLQSNasGetCellLocationInfo, 752
pack_nas_SLQSNasGetHDRColorCode, 753
pack_nas_SLQSNasGetSigInfo, 753
pack_nas_SLQSNasGetTxRxInfo, 753
pack_nas_SLQSNasIndicationRegisterExt, 754
pack_nas_SLQSNasSwiIndicationRegister, 754
pack_nas_SLQSNasSwiModemStatus, 755
pack_nas_SLQSSetBandPreference, 755
pack_nas_SLQSSetSignalStrengthsCallback, 756
pack_nas_SLQSSetSysSelectionPref, 756
pack_nas_SLQSSwiGetHDRPersonality, 756
pack_nas_SLQSSwiGetHDRProtSubtype, 757
pack_nas_SLQSSwiGetHRPDStats, 757
pack_nas_SLQSSwiGetLteCQI, 757
pack_nas_SLQSSwiGetLteSccRxInfo, 758
pack_nas_SLQSSwiNetworkDebug, 758
pack_nas_SLQSSwiPSDetach, 758
pack_nas_SetACCOLC, 747
pack_nas_SetCDMANetworkParameters, 747
pack_nas_SetLURejectCallback, 747
pack_nas_SetNetworkPreference, 748
pack_nas_SetRFInfoCallback, 748
pack_nas_SlqsGetLTECphyCAInfo, 749
unpack_nas_GetACCOLC, 759
unpack_nas_GetANAAAAuthenticationStatus, 759
unpack_nas_GetCDMANetworkParameters, 759
unpack_nas_GetHomeNetwork, 760
unpack_nas_GetHomeNetwork3GPP2, 760
unpack_nas_GetNetworkPreference, 760
unpack_nas_GetRFInfo, 760
unpack_nas_GetServingNetwork, 761
unpack_nas_GetServingNetworkCapabilities, 761
unpack_nas_GetSignalStrengths, 761
unpack_nas_InitiateDomainAttach, 762
unpack_nas_PerformNetworkScan, 762
unpack_nas_SLQSConfigSigInfo, 765
unpack_nas_SLQSGetErrorRate, 765
unpack_nas_SLQSGetNetworkTime, 765
unpack_nas_SLQSGetOperatorNameData, 766
unpack_nas_SLQSGetPLMNName, 766
unpack_nas_SLQSGetServingSystem, 766
unpack_nas_SLQSGetSignalStrength, 767
unpack_nas_SLQSGetSysInfo, 767
unpack_nas_SLQSGetSysSelectionPref, 767
unpack_nas_SLQSInitiateNetworkRegistration, 768
unpack_nas_SLQSNASSwiGetChannelLock, 771
unpack_nas_SLQSNASSwiSetChannelLock, 772
unpack_nas_SLQSNasConfigSigInfo2, 768
unpack_nas_SLQSNasGet3GPP2Subscription, 768
unpack_nas_SLQSNasGetCellLocationInfo, 768

unpack_nas_SLQSNasGetHDRColorCode, 769
unpack_nas_SLQSNasGetSigInfo, 769
unpack_nas_SLQSNasGetTxRxInfo, 769
unpack_nas_SLQSNasIndicationRegisterExt, 770
unpack_nas_SLQSNasNetworkTimeCallBack_ind, 770
unpack_nas_SLQSNasSigInfoCallback_ind, 770
unpack_nas_SLQSNasSwiIndicationRegister, 771
unpack_nas_SLQSNasSwiModemStatus, 771
unpack_nas_SLQSNasSwiOTAMessageCallback_ind, 772
unpack_nas_SLQSNasSysInfoCallback_ind, 772
unpack_nas_SLQSNasTimerCallback_ind, 773
unpack_nas_SLQSSetBandPreference, 773
unpack_nas_SLQSSetSignalStrengthsCallback, 773
unpack_nas_SLQSSetSysSelectionPref, 773
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind, 774
unpack_nas_SLQSSwiGetHDRPersonality, 774
unpack_nas_SLQSSwiGetHDRProtSubtype, 774
unpack_nas_SLQSSwiGetHRPDStats, 775
unpack_nas_SLQSSwiGetLteCQI, 775
unpack_nas_SLQSSwiGetLteSccRxInfo, 775
unpack_nas_SLQSSwiNetworkDebug, 776
unpack_nas_SLQSSwiPSDetach, 776
unpack_nas_SetACCOLC, 762
unpack_nas_SetCDMANetworkParameters, 763
unpack_nas_SetDataCapabilitiesCallback_ind, 763
unpack_nas_SetEventReportInd, 763
unpack_nas_SetLURejectCallback, 763
unpack_nas_SetNasLTECphyCaIndCallback_ind, 764
unpack_nas_SetNetworkPreference, 764
unpack_nas_SetRFInfoCallback, 764
unpack_nas_SetRoamingIndicatorCallback_ind, 764
unpack_nas_SetServingSystemCallback_ind, 765
unpack_nas_SlqsGetLTECphyCAInfo, 765
nas_ActPilotPNElement, 101
ActSetPilotPN, 101
ActSetPilotPNStrength, 101
nas_AddCDMASysInfo, 101
geoSysIdx, 102
regPrd, 102
nas_AddSysInfo, 102
cellBroadcastCap, 102
geoSysIdx, 102
nas_CDMAChannel, 104
priChA, 105
priChB, 105
secChA, 105
secChB, 105
nas_CDMAECIOThresh, 105
CDMAECIOThreshListLen, 105
pCDMAECIOThreshList, 105
nas_CDMAInfo, 105
baseId, 106
baseLat, 106
baseLong, 106
nid, 106
refpn, 106
sid, 106
nas_CDMARSSIThresh, 106
CDMARSSIThreshListLen, 107
pCDMARSSIThreshList, 107
nas_CDMASysInfo, 107
baseId, 110
baseLat, 110
baseLong, 110
bsInfoValid, 110
bsPRev, 110
bsPRevValid, 110
ccsSupported, 110
ccsSupportedValid, 110
cdmaSysIdValid, 110
isSysPrlMatch, 110
isSysPrlMatchValid, 110
MCC, 110
MNC, 110
networkID, 110
networkIdValid, 110
pRevInUse, 110
pRevInUseValid, 111
packetZone, 110
packetZoneValid, 110
sysInfoCDMA, 111
systemID, 111
nas_CDMASysInfoExt, 111
imsi_11_12, 111
MCC, 111
nas_CSGID, 114
id, 114
mcc, 115
mnc, 115
mncPcsDigits, 115
rat, 115
nas_CallBarringSysInfo, 102
csBarStatus, 103
psBarStatus, 103
nas_CommInfo, 112
imsRegState, 114
modemMode, 114
psState, 114
systemMode, 114
temperature, 114
nas_DRCParams, 121
DRCCover, 122
DRCValue, 122
nas_DataStatusDetail, 116
IPAddress, 118
LastErrCode, 118
nas_DeviceConfigDetail, 120
Chipset, 121
HWVersion, 121

QLIC, 121
Technology, 121
nas_ECIOThresh, 123
ECIOThresListLen, 123
pECIOThresList, 123
nas_GERANInfo, 124
arfcn, 126
bsic, 126
cellID, 126
insNmrCellInfo, 126
lac, 126
nmrInst, 126
plmn, 126
rxLev, 126
timingAdvance, 126
nas_GSMRSSIThresh, 128
GSMRSSIThreshListLen, 128
pGSMRSSIThreshList, 128
nas_GSMSrvStatusInfo, 128
isPrefDataPath, 129
srvStatus, 129
trueSrvStatus, 129
nas_GSMSysInfo, 129
cellId, 131
cellIdValid, 131
dtmSupp, 131
dtmSuppValid, 131
egprsSupp, 132
egprsSuppValid, 132
lac, 132
lacValid, 132
MCC, 132
MNC, 132
networkIdValid, 132
regRejectInfoValid, 132
rejCause, 132
rejectSrvDomain, 132
sysInfoGSM, 132
nas_HDRECIOThresh, 132
HDRECIOThreshListLen, 132
pHDRECIOThreshList, 132
nas_HDRIOThresh, 132
HDRIOThreshListLen, 133
pHDRIOThreshList, 133
nas_HDRRSSIThresh, 133
HDRRSSIThreshListLen, 133
pHDRRSSIThreshList, 133
nas_HDRSINRThresh, 134
HDRSINRThresListLen, 134
pHDRSINRThresList, 134
nas_HDRSINRThreshold, 134
HDRSINRThreshListLen, 135
pHDRSINRThreshList, 135
nas_HDRSysInfo, 135
hdrActiveProt, 137
hdrActiveProtValid, 137
hdrPersonality, 137
hdrPersonalityValid, 137
is856SysId, 137
is856SysIdValid, 137
isSysPrlMatch, 137
isSysPrlMatchValid, 137
sysInfoHDR, 137
nas_IOThresh, 139
IOThresListLen, 139
pIOThresList, 139
nas_LTEInfo, 141
band, 143
bandwidth, 143
emmConnState, 143
emmState, 143
emmSubState, 143
RXChan, 143
TXChan, 143
nas_LTEInfoInterfreq, 144
freqsLen, 144
InfoInterfreq, 144
ueInIdle, 144
nas_LTEInfoIntrafreq, 144
CellParams, 146
cellReselPriority, 146
cellsLen, 146
earfcn, 146
globalCellId, 146
plmn, 146
sIntraSearch, 146
sNonIntraSearch, 146
servingCellId, 146
tac, 146
threshServingLow, 146
ueInIdle, 146
nas_LTEInfoNeighboringGSM, 146
freqsLen, 147
LteGsmCellInfo, 147
ueInIdle, 147
nas_LTEInfoNeighboringWCDMA, 147
freqsLen, 148
ueInIdle, 148
nas_LTERSRPThresh, 149
LTERSRPThreshListLen, 149
pLTERSRPThreshList, 149
nas_LTERSRQThresh, 149
LTERSRQThreshListLen, 150
pLTERSRQThreshList, 150
nas_LTERSSIThresh, 150
LTERSSIThreshListLen, 150
pLTERSSIThreshList, 150
nas_LTESNRThresh, 153
LTESNRThresListLen, 153
pLTESNRThresList, 153
nas_LTESNRThreshold, 153
LTESNRThreshListLen, 154
pLTESNRThreshList, 154
nas_LTESigRptCfg, 150
avgPeriod, 151
rptRate, 151

nas_LTESigRptConfig, 151
avgPeriod, 152
rptRate, 152
nas_LTESysInfo, 154
cellId, 156
cellIdValid, 156
lac, 156
lacValid, 156
MCC, 156
MNC, 156
networkIdValid, 156
regRejectInfoValid, 156
rejCause, 156
rejectSrvDomain, 156
sysInfoLTE, 156
tac, 156
tacValid, 156
nas_MNRInfo, 158
mcc, 159
mnc, 159
rat, 159
nas_NetworkStat1x, 160
ActSetCnt, 162
NeighborSetCnt, 162
pActPilotPNElements, 162
pNeighborSetPilotPN, 162
RX_EC_IO, 162
RX_PWR, 162
SO, 162
State, 162
TX_PWR, 162
nas_NetworkStatEVDO, 162
MACIndex, 163
PER, 164
pSectorID, 164
PilotEnergy, 164
RX_PWR, 164
SNR, 164
SectorIDLen, 164
State, 164
nas_OperatorPLMNData, 165
lac1, 166
lac2, 166
mcc, 166
mnc, 166
PLMNRecID, 166
nas_PLMNNetworkName, 173
numInstance, 173
PLMNNetName, 173
nas_PLMNNetworkNameData, 174
codingScheme, 175
countryInitials, 175
longName, 175
longNameLen, 175
longNameSpareBits, 175
shortName, 175
shortNameLen, 175
shortNameSpareBits, 175
nas_PhyCaAggPcellInfo, 167
dl_bw_value, 167
freq, 167
iLTEbandValue, 167
pci, 168
TlvPresent, 168
nas_PhyCaAggScellDlBw, 168
dl_bw_value, 168
TlvPresent, 168
nas_PhyCaAggScellIndType, 169
freq, 169
pci, 169
scell_state, 169
TlvPresent, 169
nas_PhyCaAggScellIndex, 168
scell_idx, 169
TlvPresent, 169
nas_PhyCaAggScellInfo, 169
dl_bw_value, 172
freq, 172
iLTEbandValue, 172
pci, 172
scell_state, 172
TlvPresent, 172
nas_PilotSetData, 172
NumPilots, 172
pPilotSetInfo, 172
nas_PilotSetParams, 172
PilotPN, 173
PilotStrength, 173
PilotType, 173
nas_QmiNas3GppNetworkInfo, 177
Desription, 178
Forbidden, 178
InUse, 178
MCC, 178
MNC, 178
Preferred, 178
Roaming, 178
nas_QmiNas3GppNetworkRAT, 178
MCC, 179
MNC, 179
RAT, 179
nas_QmisNasPcsDigit, 179
includes_pcs_digit, 179
MCC, 179
MNC, 179
nas_RFInfoTlv, 180
activeBandClass, 180
activeChannel, 180
radioInterface, 180
radioInterfaceSize, 180
TlvPresent, 180
nas_RSRPThresh, 181
pRSRPThresList, 182
RSRPThresListLen, 182
nas_RSRQThresh, 183
pRSRQThresList, 183

RSRQThresListLen, 183
nas_RSSIThresh, 183
pRSSIThresList, 184
RSSIThresListLen, 184
nas_RejectReasonTlv, 180
rejectCause, 180
serviceDomain, 180
TlvPresent, 180
nas_RxSigInfo, 185
isRadioTuned, 186
rsrp, 186
rxChainIndex, 186
rxPower, 186
nas_SLQSSignalStrengthsIndReq, 191
ecioDelta, 191
ecioThresholdList, 191
ecioThresholdListLen, 192
ioDelta, 192
lteRsrpDelta, 192
lteSnrDelta, 192
rsrqDelta, 192
rxSignalStrengthDelta, 192
sinrDelta, 192
sinrThresholdList, 192
sinrThresholdListLen, 192
nas_SLQSSignalStrengthsInformation, 192
ecioInfo, 192
errorRateInfo, 192
io, 192
lteRsrpinfo, 192
lteSnrinfo, 193
rsrqInfo, 193
rxSignalStrengthInfo, 193
sinr, 193
nas_SLQSSignalStrengthsTlv, 193
sSLQSSignalStrengthsInfo, 193
TlvPresent, 193
nas_SccRxInfo, 187
numInstances, 187
rsrq, 187
sigInfo, 188
snr, 188
TlvPresent, 188
nas_SignalStrengthTlv, 190
radioInterface, 191
signalStrength, 191
TlvPresent, 191
nas_SrvStatusInfo, 193
isPrefDataPath, 194
srvStatus, 194
nas_TDSCDMAECIOThresh, 196
nas_TDSCDMARSCPThresh, 197
nas_TDSCDMARSSIThresh, 197
nas_TDSCDMASINRCONFThresh, 198
nas_TDSCDMASINRThresh, 198
nas_UMTSInfo, 202
cellID, 203
ecio, 203
geranInst, 203
GeranInstInfo, 203
lac, 203
plmn, 203
psc, 203
rscp, 203
UMTSInstInfo, 204
uarfcn, 203
umtsInst, 203
nas_UMTSinstInfo, 204
umtsEcio, 204
umtsPsc, 204
umtsRscp, 204
umtsUarfcn, 204
nas_UniversalTime, 205
day, 206
dayOfWeek, 206
hour, 206
minute, 206
month, 206
second, 206
year, 206
nas_WCDMAECIOThresh, 207
nas_WCDMAInfoLTENeighborCell, 208
umtsLTENbrCellLen, 209
wcdmaRRCState, 209
nas_WCDMARSSIThresh, 209
nas_WCDMASysInfo, 209
cellId, 212
cellIdValid, 212
hsCallStatus, 212
hsCallStatusValid, 212
hsInd, 212
hsIndValid, 212
lac, 212
lacValid, 212
MCC, 212
MNC, 212
networkIdValid, 212
psc, 212
pscValid, 212
regRejectInfoValid, 212
rejCause, 213
rejectSrvDomain, 213
sysInfoWCDMA, 213
nas_acqOrderPref, 100
acqOrdeLen, 101
pAcqOrder, 101
nas_callBarStatus, 103
csBarStatus, 104
psBarStatus, 104
nas_cellParams, 111
pci, 112
rsrp, 112
rsrq, 112
rssi, 112
srxlev, 112
nas_currentPLMN, 115

MCC, 115
MNC, 115
netDescr, 115
netDescrLength, 116
nas_dataSrvCapabilities, 116
dataCapabilities, 116
dataCapabilitiesLen, 116
nas_detailSvcInfo, 119
hdrHybrid, 120
hdrSrvStatus, 120
isSysForbidden, 120
srvCapability, 120
srvStatus, 120
nas_dirNum, 121
dirNum, 121
dirNumLen, 121
nas_ecioListElement, 122
ecio, 122
radioIf, 122
nas_errorRateListElement, 123
errorRate, 124
radioIf, 124
nas_geranInstInfo, 126
geranArfcn, 126
geranBsicBcc, 126
geranBsicNcc, 127
geranRssi, 127
nas_gsmCellInfo, 127
arfcn, 127
band1900, 128
bsicId, 128
cellIdValid, 128
rssi, 128
srxlev, 128
nas_homeSIDNID, 137
numInstances, 137
SidNid, 137
nas_infoInterFreq, 138
cell_resel_priority, 138
cellInterFreqParams, 138
cells_len, 138
earfcn, 138
threshXHigh, 139
threshXLow, 139
nas_lteEARFCN, 139
earfcn0, 140
earfcn1, 140
status, 140
nas_lteGsmCellInfo, 140
cellReselPriority, 141
cells_len, 141
GsmCellInfo, 141
nccPermitted, 141
threshGsmHigh, 141
threshGsmLow, 141
nas_ltePCI, 148
earfcn, 148
pci, 148
status, 148
nas_lteRsrpinformation, 148
rsrplevel, 149
nas_lteSnrinformation, 152
snrlevel, 152
nas_lteWcdmaCellInfo, 156
cellReselPriority, 157
cellsLen, 157
threshXhigh, 157
threshXlow, 157
uarfcn, 157
WCDMACellInfo, 157
nas_major
NASLteNasReleaseInfoTlv, 215
nas_minBasedIMSI, 157
imsiM1112, 158
imsiMS1, 158
imsiMS2, 158
mccM, 158
nas_minor
NASLteNasReleaseInfoTlv, 215
nas_namName, 159
namName, 159
namNameLen, 159
nas_netSelectionPref, 159
mcc, 160
mnc, 160
netReg, 160
nas_nmrCellInfo, 164
nmrArfcn, 165
nmrBsic, 165
nmrCellID, 165
nmrLac, 165
nmrPlmn, 165
nmrRxLev, 165
nas_operatorNameString, 165
PLMNName, 165
nas_operatorPLMNList, 166
numInstance, 167
PLMNData, 167
nas_protocolSubtypeElement, 175
AccessMac, 176
AuthProt, 176
ControlMac, 176
EncryptProt, 176
ForwardMac, 176
IdleState, 176
KeyExchange, 176
MultDisc, 176
PhysicalLayer, 177
ReverseMac, 177
SecProt, 177
VirtStream, 177
nas_qaQmi3Gpp2TimeZone, 177
daylightSavings, 177
leapSeconds, 177
localTimeOffset, 177
nas_release

NASLteNasReleaseInfoTlv, 215
nas_roamIndList, 181
numInstances, 181
radioInterface, 181
roamIndicator, 181
nas_rsrqInformation, 182
radioIf, 182
rsrq, 183
nas_rxInfo, 184
ecio, 185
isRadioTuned, 185
phase, 185
rscp, 185
rsrp, 185
rxPower, 185
nas_rxSignalStrengthListElement, 186
radioIf, 187
rxSignalStrength, 187
nas_servSystem, 188
csAttachState, 190
numRadioInterfaces, 190
psAttachState, 190
radioInterface, 190
regState, 190
selNetwork, 190
nas_serviceProviderName, 188
displayCondition, 188
spn, 188
spnLength, 188
nas_sidNid, 190
nid, 190
sid, 190
nas_sysInfoCommon, 194
isSysForbidden, 196
isSysForbiddenValid, 196
roamStatus, 196
roamStatusValid, 196
srvCapability, 196
srvCapabilityValid, 196
srvDomain, 196
srvDomainValid, 196
nas_timeInfo, 199
day, 200
dayLtSavingAdj, 200
dayOfWeek, 200
hour, 200
minute, 200
month, 200
radioInterface, 200
second, 200
timeZone, 200
TlvPresent, 200
year, 200
nas_trueIMSI, 200
imsiT1112, 201
imsiTS1, 201
imsiTS2, 201
imsiTaddrNum, 201
mccT, 201
nas_txInfo, 201
isInTraffic, 202
txPower, 202
nas_umtsLTENbrCell, 204
cellIsTDD, 205
earfcn, 205
pci, 205
rsrp, 205
rsrq, 205
srxlev, 205
nas_wcdmaCellInfo, 206
cpich_ecno, 207
cpich_rscp, 207
psc, 207
srxlev, 207
nas_wcdmaUARFCN, 213
status, 213
uarfcn, 213
NasGetLTECphyCaInfo, 214
PhyCaAggPcellInfo, 214
PhyCaAggScellDlBw, 214
PhyCaAggScellIndType, 214
PhyCaAggScellIndex, 214
PhyCaAggScellInfo, 214
nasRelInfoTlv
NASQmiCbkNasSwiOTAMessageInd, 220
nccPermitted
nas_lteGsmCellInfo, 141
NeighborSetCnt
nas_NetworkStat1x, 162
netDescr
nas_currentPLMN, 115
netDescrLength
nas_currentPLMN, 116
netInfoLen
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
netReg
nas_netSelectionPref, 160
NetSelPref
NASNetSelPreferenceTlv, 215
NetworkID
unpack_nas_SLQSGetServingSystem_t, 471
networkID
nas_CDMASysInfo, 110
networkIdValid
nas_CDMASysInfo, 110
nas_GSMSysInfo, 132
nas_LTESysInfo, 156
nas_WCDMASysInfo, 212
networkInfoLen
unpack_wds_SLQSGetCurrDataSystemStat_t, 586
NetworkType
currNetworkInfo, 48
wds_currNetworkInfo, 636
NewMMTlv

unpack_sms_SetNewSMSCallback_ind_t, 523
newMTMessageTlv, 223
MTMessageInfo, 223
TlvPresent, 223
newPINLen
uim_unblockUIMPIN, 399
newPINVal
uim_unblockUIMPIN, 399
newPasswd
pack_voice_SLQSVoiceSetCallBarringPassword_t, 323
newPasswdAgain
pack_voice_SLQSVoiceSetCallBarringPassword_t, 323
newPin
pack_dms_UIMUnblockPIN_t, 235
newValue
pack_dms_UIMChangePIN_t, 231
nextHeader
LibPackTFTIDParams, 84
nid
nas_CDMAInfo, 106
nas_sidNid, 190
unpack_nas_GetHomeNetwork_t, 457
nmrArfcn
nas_nmrCellInfo, 165
nmrBsic
nas_nmrCellInfo, 165
nmrCellID
nas_nmrCellInfo, 165
nmrInst
nas_GERANInfo, 126
nmrLac
nas_nmrCellInfo, 165
nmrPlmn
nas_nmrCellInfo, 165
nmrRxLev
nas_nmrCellInfo, 165
noReplyTimer
voice_callFWExtInfo, 612
voice_callFWInfo, 613
notification
unpack_omaDmNotificationsTlv_t, 497
notificationType
sMSEtwsMessage, 370
notused
unpack_dms_SetCrashAction_t, 413
num_instances
DMScustSettingList, 51
sms_qaQmi3GPP2BroadcastCfgInfo, 362
sms_qaQmi3GPPBroadcastCfgInfo, 363
numApp
slotInf, 355
uim_slotInfo, 398
numCrashes
crashInformation, 48
numEntries
unpack_dms_SLQSSwiGetFirmwareCurr_t, 424
numFeatures
uim_personalizationStatus, 390
numFiles
uim_registerRefresh, 394
NumFlows
unpack_qos_SLQSSetQosEventCallback_ind_t, 507
numInstance
nas_operatorPLMNList, 167
nas_PLMNNetworkName, 173
numInstances
nas_homeSIDNID, 137
nas_roamIndList, 181
nas_SccRxInfo, 187
sms_getMsgWaitingInfo, 358
voice_arrAlertingPattern, 601
voice_arrAlertingType, 602
voice_arrAlphaID, 603
voice_arrCalledPartyNum, 603
voice_arrCallEndReason, 604
voice_arrCallInfo, 604
voice_arrConnectPartyNum, 605
voice_arrDiagInfo, 605
voice_arrRedirPartyNum, 606
voice_arrRemotePartyName, 606
voice_arrRemotePartyNum, 607
voice_arrSvcOption, 608
voice_arrUUSInfo, 608
voice_getCallFWExtInfo, 626
voice_getCallFWInfo, 627
wds_DomainNameList, 641
wds_PCSCFFQDNAddressList, 644
wds_PCSCFIPv4ServerAddressList, 645
numLen
voice_callFWExtInfo, 612
voice_callFWInfo, 614
voice_peerNumberInfo, 628
voice_remotePartyNum, 632
numOfFiles
uim_refreshevent, 393
numOfRoutes
sms_setRoutesReq, 368
numOpt
wds_DHCPLeaseOptTlv, 637
wds_DHCPv4OptionList, 640
wdsDhcpv4OptionList, 650
numPI
voice_peerNumberInfo, 628
NumPilots
nas_PilotSetData, 172
numPlan
voice_callFWExtInfo, 613
voice_connectNumInfo, 621
voice_peerNumberInfo, 628
numPresInd
voice_connectNumInfo, 621
numQosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 507

numRadioInterfaces
nas_servSystem, 190
NumRxFilters
unpack_qos_QosFlowInfo_t, 503
numSI
voice_peerNumberInfo, 628
numSlot
uim_cardStatus, 382
NumSupUSBComps
unpack_dms_GetUSBComp_t, 412
NumTxFilters
unpack_qos_QosFlowInfo_t, 503
numType
voice_callFWExtInfo, 613
voice_connectNumInfo, 621
voice_peerNumberInfo, 628
number
voice_callFWExtInfo, 612
voice_callFWInfo, 613
voice_peerNumberInfo, 628
numberPlan
voice_callFwdTypeAndPlan, 611
numberType
voice_callFwdTypeAndPlan, 611
NxtHdrProto
unpack_qos_swiQosFilter_t, 513

OKtoRefresh
pack_uim_SLQSUIMRefreshOK_t, 306
OMADMEnabled
unpack_swiavms_SLQSAvMSGetSettings_t, 530
unpack_swioma_SLQSOMADMGetSettings_t, 538
oddEvenInd
voice_calledPartySubAdd, 610
OfflineReason
unpack_dms_GetPower_t, 411
offset
uim_readTransparentInfo, 391
oldPINLen
uim_changeUIMPIN, 383
oldPINVal
uim_changeUIMPIN, 383
oldPasswd
pack_voice_SLQSVoiceSetCallBarringPassword_t, 323
oldValue
pack_dms_UIMChangePIN_t, 231
opaqueId
pack_loc_SLQSLOCInjectSensorData_t, 248
operatingMode
dms_OperatingModeTlv, 50
OperatingModeTlv
unpack_dms_SetEventReport_ind_t, 415
operation
pack_pds_SetPDSDefaults_t, 276
uim_depersonalizationInformation, 384
OperationMode
unpack_dms_GetPower_t, 411
optCode
wds_DHCPOpt, 638
wds_DHCPv4Option, 639
wdsDhcpv4Option, 650
optList
wds_DHCPLeaseOptTlv, 637
optListData
wds_DHCPLeaseOptTlv, 637
optVal
wds_DHCPv4Option, 639
wdsDhcpv4Option, 650
optValLen
wds_DHCPOpt, 638
wds_DHCPv4Option, 639
wdsDhcpv4Option, 650
otaMsgTlv
NASQmiCbkNasSwiOTAMessageInd, 220

p1Status
unpack_dms_UIMGetPINStatus_t, 432
p1UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 432
p1VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 432
p2Status
unpack_dms_UIMGetPINStatus_t, 432
p2UnblockRetriesLeft
unpack_dms_UIMGetPINStatus_t, 432
p2VerifyRetriesLeft
unpack_dms_UIMGetPINStatus_t, 432
p3GPP2TimeInfo
unpack_nas_SLQSGetNetworkTime_t, 467
p3GPPTimeInfo
unpack_nas_SLQSGetNetworkTime_t, 467
p3GppNetworkInfoInstances
unpack_nas_PerformNetworkScan_t, 461
p3GppNetworkInstanceSize
unpack_nas_PerformNetworkScan_t, 461
p3gppRelease
pack_wds_SLQSSet3GPPConfigItem_t, 341
pAAASPI
pack_wds_SetMobileIPProfile_t, 335
PACK_WDS_IPV4
wds.h, 939
PACK_WDS_IPV6
wds.h, 939
pAMRStatus
pack_voice_SLQSVoiceGetConfig_t, 319
pAPN
_PackSwiAVMSSettingsAPNInfo, 29
pAPNClass
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pAPNClass3GPP2
LibpackProfile3GPP2, 69
LibPackprofile_3GPP2, 80
pAPNDisabledFlag
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pAPNEnabled3GPP2

LibpackProfile3GPP2, 69
LibPackprofile_3GPP2, 80
pAPNInfo
pack_swiavms_SLQSAvMSSetSettings_t, 291
unpack_swiavms_SLQSAvMSGetSettings_t, 530
pAPNName
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pAPNnameSize
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pAccuracy
unpack_pds_GetPDSDefaults_t, 497
pAcqOrder
nas_acqOrderPref, 101
pAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 274
pActPilotPNElements
nas_NetworkStat1x, 162
pAddCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pAddGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pAddHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pAddLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pAddWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pAddrAllocPref
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pAddress
pack_wds_SetMobileIPProfile_t, 335
pAirTimer
pack_voice_SLQSVoiceGetConfig_t, 319
pAirTimerCnt
unpack_voice_SLQSVoiceGetConfig_t, 567
pAirTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 324
pAirTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 571
pAlertType
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pAlertingPattern
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pAllowLinger
LibpackProfile3GPP2, 69
LibPackprofile_3GPP2, 80
pAlphaIDInfo
unpack_voice_SLQSOriginateUSSD_t, 548
unpack_voice_SLQSVoiceDialCall_t, 550
unpack_voice_SLQSVoiceGetCallBarring_t, 554
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 556
unpack_voice_SLQSVoiceGetCallInfo_t, 558
unpack_voice_SLQSVoiceGetCallWaiting_t, 559
unpack_voice_SLQSVoiceGetCLIP_t, 561
unpack_voice_SLQSVoiceGetCLIR_t, 562
unpack_voice_SLQSVoiceGetCNAP_t, 563
unpack_voice_SLQSVoiceGetCOLP_t, 564
unpack_voice_SLQSVoiceGetCOLR_t, 566
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 569
unpack_voice_SLQSVoiceSetSUPSService_t, 572
pAltitudeAssumed
unpack_loc_PositionRpt_Ind_t, 449
pAltitudeWrtEllipsoid
unpack_loc_BestAvailPos_Ind_t, 440
unpack_loc_PositionRpt_Ind_t, 449
pAltitudeWrtMeanSeaLevel
unpack_loc_BestAvailPos_Ind_t, 440
unpack_loc_PositionRpt_Ind_t, 449
pApnString
LibpackProfile3GPP2, 69
LibPackprofile_3GPP2, 81
pApnStringSize
LibpackProfile3GPP2, 69
LibPackprofile_3GPP2, 81
pApnname
pack_wds_SetDefaultProfile_t, 332
pAppName
loc_LocApplicationInfo, 93
pAppPriority
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pAppProvider
loc_LocApplicationInfo, 93
pAppSubType
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 488
pAppType
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pAppVersion
loc_LocApplicationInfo, 93
pApplication
pack_nas_SetCDMANetworkParameters_t, 253
pApplicationInfo
pack_loc_Start_t, 250
pArrAlertingPattern
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrAlertingType
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrAlphaID
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrCallEndReason
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrCallInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrCalledPartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553

pArrConnectPartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrDiagInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrRedirPartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrRemotePartyName
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrRemotePartyNum
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrSvcOption
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pArrUUSInfo
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
pAuth
pack_wds_SLQSStartDataSession_t, 345
pAuthPassword
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pAuthPassword_tSize
LibPackprofile_3GPP2, 81
pAuthPasswordSize
LibpackProfile3GPP2, 70
pAuthProtocol
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pAuthRetryCount
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pAuthTimeout
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pAuthenticateResult
unpack_uim_SLQSUIMAuthenticate_t, 543
pAuthenticationPref
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pAutoAnsStatus
unpack_voice_SLQSVoiceSetConfig_t, 571
pAutoAnswer
pack_voice_SLQSVoiceGetConfig_t, 319
pack_voice_SLQSVoiceSetConfig_t, 324
pAutoAnswerStat
unpack_voice_SLQSVoiceGetConfig_t, 568
pAutoSelection
unpack_uim_SLQSUIMGetConfiguration_t, 545
pAutosdm
pack_swioma_SLQSOMADMSetSettings_t, 297
pBandPref
pack_nas_SLQSSetSysSelectionPref_t, 274
unpack_nas_SLQSGetSysSelectionPref_t, 478
pBdsSVInfo
pack_loc_Delete_Assist_Data_t, 238
pBinaryUpdateSessionInfo
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
_unpack_swiavms_SLQSAvMSSessionGetInfo_t, 32
pBroadcast
pack_nas_SetCDMANetworkParameters_t, 253
pBroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 288
unpack_sms_SLQSGetSmsBroadcastConfig_t, 526
pBurstDTMFLengths
pack_voice_SLQSVoiceBurstDTMF_t, 313
pCCResType
unpack_voice_SLQSVoiceGetCallBarring_t, 554
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 556
unpack_voice_SLQSVoiceGetCallWaiting_t, 560
unpack_voice_SLQSVoiceGetCLIP_t, 561
unpack_voice_SLQSVoiceGetCLIR_t, 562
unpack_voice_SLQSVoiceGetCNAP_t, 563
unpack_voice_SLQSVoiceGetCOLP_t, 565
unpack_voice_SLQSVoiceGetCOLR_t, 566
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 570
pCCResultType
unpack_voice_SLQSVoiceDialCall_t, 550
unpack_voice_SLQSVoiceSetSUPSService_t, 572
pCCSUPSType
unpack_voice_SLQSVoiceDialCall_t, 550
unpack_voice_SLQSVoiceGetCallBarring_t, 554
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 556
unpack_voice_SLQSVoiceGetCallWaiting_t, 560
unpack_voice_SLQSVoiceGetCLIP_t, 561
unpack_voice_SLQSVoiceGetCLIR_t, 562
unpack_voice_SLQSVoiceGetCNAP_t, 563
unpack_voice_SLQSVoiceGetCOLP_t, 565
unpack_voice_SLQSVoiceGetCOLR_t, 566
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 570
unpack_voice_SLQSVoiceSetSUPSService_t, 572
pCCSuppsType
unpack_voice_SLQSOriginateUSSD_t, 548
pCDMABroadcastConfig
pack_sms_SLQSSetSmsBroadcastConfig_t, 288
unpack_sms_SLQSGetSmsBroadcastConfig_t, 526
pCDMAChannel
unpack_nas_SLQSNasGet3GPP2Subscription_t, 479
pCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 261
pCDMAECIOThreshList
nas_CDMAECIOThresh, 105
pCDMAFrameErrRate
unpack_nas_SLQSGetErrorRate_t, 466
pCDMAInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 261

pCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 261
pCDMARSSIThreshList
nas_CDMARSSIThresh, 107
pCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 484
pCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pCLIPResp
unpack_voice_SLQSVoiceGetCLIP_t, 561
pCLIRResp
unpack_voice_SLQSVoiceGetCLIR_t, 562
pCLIRType
pack_voice_SLQSVoiceDialCall_t, 314
pCNAPResp
unpack_voice_SLQSVoiceGetCNAP_t, 563
pCOLPResp
unpack_voice_SLQSVoiceGetCOLP_t, 565
pCOLRResp
unpack_voice_SLQSVoiceGetCOLR_t, 566
PCSCFAddrPCO
unpack_wds_SLQSGetRuntimeSettings_t, 590
PCSCFFQDNAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 590
pCSGID
pack_nas_SLQSSetSysSelectionPref_t, 274
pCUGInfo
pack_voice_SLQSVoiceDialCall_t, 314
pCallBarringPasswd
pack_voice_SLQSVoiceSetSUPSService_t, 327
pCallForwardingNumber
pack_voice_SLQSVoiceSetSUPSService_t, 327
pCallFwdTypeAndPlan
pack_voice_SLQSVoiceSetSUPSService_t, 327
pCallID
pack_voice_SLQSVoiceManageCalls_t, 320
pack_voice_SLQSVoiceSendFlash_t, 322
pack_voice_SLQSVoiceStartContDTMF_t, 327
unpack_voice_SLQSVoiceBurstDTMF_t, 549
unpack_voice_SLQSVoiceDialCall_t, 550
unpack_voice_SLQSVoiceGetCallBarring_t, 554
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 556
unpack_voice_SLQSVoiceGetCallWaiting_t, 560
unpack_voice_SLQSVoiceGetCLIP_t, 561
unpack_voice_SLQSVoiceGetCLIR_t, 562
unpack_voice_SLQSVoiceGetCNAP_t, 563
unpack_voice_SLQSVoiceGetCOLP_t, 564
unpack_voice_SLQSVoiceGetCOLR_t, 566
unpack_voice_SLQSVoiceSendFlash_t, 569
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 570
unpack_voice_SLQSVoiceSetSUPSService_t, 572
unpack_voice_SLQSVoiceStartContDTMF_t, 573
voice_burstDTMFInfo, 609
pCallId
pack_voice_SLQSVoiceAnswerCall_t, 312
pack_voice_SLQSVoiceEndCall_t, 315
unpack_voice_SLQSOriginateUSSD_t, 548
unpack_voice_SLQSVoiceAnswerCall_t, 549
unpack_voice_SLQSVoiceEndCall_t, 550
pCallInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pCallPartySubAdd
pack_voice_SLQSVoiceDialCall_t, 314
pCallType
pack_voice_SLQSVoiceDialCall_t, 314
pCardResult
unpack_uim_ReadTransparent_t, 541
unpack_uim_SLQSUIMAuthenticate_t, 543
unpack_uim_SLQSUIMGetFileAttributes_t, 545
pCardStatus
unpack_uim_GetCardStatus_t, 540
unpack_uim_SLQSUIMSetStatusChangeCall-Back_ind_t, 547
pCcResultType
unpack_voice_SLQSOriginateUSSD_t, 548
pCellDb
pack_loc_Delete_Assist_Data_t, 238
pChangeDuration
pack_nas_SLQSInitiateNetworkRegistration_t, 257
pChgDuration
pack_nas_SLQSSetSysSelectionPref_t, 274
pClientPerformOperationFlag
pack_swiavms_SLQSAvMSSendSelection_t, 290
pClkInfo
pack_loc_Delete_Assist_Data_t, 238
pColorCode
unpack_nas_SLQSNasGetHDRColorCode_t, 481
pConfig
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
_unpack_swiavms_SLQSAvMSSessionGetInfo_t, 32
pConfigAltitudeAssumed
pack_loc_Start_t, 251
pConfigurationMask
pack_uim_SLQSUIMGetConfiguration_t, 302
pConnectNumInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pConnectionRequest
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
pConnectionRetryTimers
pack_swiavms_SLQSAvMSSetSettings_t, 291
unpack_swiavms_SLQSAvMSGetSettings_t, 530
pCreateProfileOut
unpack_wds_SLQSCreateProfile_t, 583
pCurAMRConfig
unpack_voice_SLQSVoiceGetConfig_t, 568
pCurPrefVoiceSO
unpack_voice_SLQSVoiceGetConfig_t, 568

pCurProfile
pack_wds_SLQSCreateProfile_t, 336
pCurVoiceDomainPref
unpack_voice_SLQSVoiceGetConfig_t, 568
pCurVoicePrivacyPref
unpack_voice_SLQSVoiceGetConfig_t, 568
pCurrChannelRateInd
pack_wds_SLQSWdsSetEventReport_t, 347
pCurrDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 347
pCurrImgInfo
unpack_dms_SLQSSwiGetFirmwareCurr_t, 424
pCurrPrefDataSysInd
pack_wds_SLQSWdsSetEventReport_t, 347
pCurrTTYMode
unpack_voice_SLQSVoiceGetConfig_t, 568
pCurrentPersonality
unpack_nas_SLQSSwiGetHDRPersonality_t, 487
pCurrentPrsnlty
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 488
pCustSettingInfo
DMSgetCustomFeatureV2, 52
pCustSettingList
DMSgetCustomFeatureV2, 52
pCustomSCP
pack_nas_SetCDMANetworkParameters_t, 253
pDDTMInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
PDOP
loc_precisionDilution, 94
PDPType
unpack_wds_SLQSGetRuntimeSettings_t, 590
pDRCParams
unpack_nas_SLQSSwiGetHRPDStats_t, 488
pDataBearer
unpack_wds_GetDataBearerTechnology_t, 576
pDataBearerTechInd
pack_wds_SLQSWdsSetEventReport_t, 347
pDataCallStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 347
pDataMode
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pDataRate
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pDataSessionStatus
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
pDataStatusDetail
unpack_nas_SLQSSwiNetworkDebug_t, 490
pDataSystemStatusChangeInd
pack_wds_SLQSWdsSetEventReport_t, 347
pDayltSavAdj
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 483
pDefaultPDNEnabled
pack_wds_SLQSSet3GPPConfigItem_t, 341
pDeferTime
pack_swiavms_SLQSAvMSSendSelection_t, 290
pack_swioma_SLQSOMADMSendSelection_t, 296
pDestSMSContent
pack_dms_SLQSSwiSetDyingGaspCfg_t, 229
packgetDyingGaspCfg, 349
pDestSMSNum
pack_dms_SLQSSwiSetDyingGaspCfg_t, 229
packgetDyingGaspCfg, 349
pDetachAction
pack_nas_SLQSSwiPSDetach_t, 275
pDeviceConfigDetail
unpack_nas_SLQSSwiNetworkDebug_t, 490
pDiagInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pDigitBuff
voice_burstDTMFInfo, 609
pDirNum
unpack_nas_SLQSNasGet3GPP2Subscription_t, 479
pDormancyStatusInd
pack_wds_SLQSWdsSetEventReport_t, 347
pDualStandByPrefInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pECIOThresList
nas_ECIOThresh, 123
pECIOThresh
pack_nas_SLQSConfigSigInfo_t, 256
PER
nas_NetworkStatEVDO, 164
pEVDOPageMonPerChangeInd
pack_wds_SLQSWdsSetEventReport_t, 347
pEmerMode
pack_nas_SLQSSetSysSelectionPref_t, 274
unpack_nas_SLQSGetSysSelectionPref_t, 478
pEmergencyCategory
pack_voice_SLQSVoiceDialCall_t, 314
pEnableNotification
pack_wds_DHCPv4ClientLeaseChange_t, 328
pEnabled
pack_wds_SetMobileIPProfile_t, 335
pEnabledStatus
unpack_pds_GetPDSState_t, 498
pEncryptData
pack_uim_ReadTransparent_t, 299
pEncryptedData
unpack_uim_ReadTransparent_t, 541
pEncryptedPIN1
pack_uim_VerifyPin_t, 309
unpack_uim_ChangePin_t, 539
unpack_uim_SetPinProtection_t, 542
unpack_uim_UnblockPin_t, 547
unpack_uim_VerifyPin_t, 548
pErrorRateInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pExtErrCode
UnPackGetProfileSettingOut, 598

pExtErrorCode
unpack_wds_SLQSModifyProfile_t, 591
pFailCause
unpack_voice_SLQSVoiceGetCallBarring_t, 554
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 556
unpack_voice_SLQSVoiceGetCallWaiting_t, 560
unpack_voice_SLQSVoiceGetCLIP_t, 561
unpack_voice_SLQSVoiceGetCLIR_t, 562
unpack_voice_SLQSVoiceGetCNAP_t, 563
unpack_voice_SLQSVoiceGetCOLP_t, 565
unpack_voice_SLQSVoiceGetCOLR_t, 566
unpack_voice_SLQSVoiceManageCalls_t, 568
unpack_voice_SLQSVoiceSetCallBarringPassword-_t, 570
unpack_voice_SLQSVoiceSetSUPSService_t, 572
pFailureReason
unpack_wds_SLQSStartDataSession_t, 596
pFileAttributes
unpack_uim_SLQSUIMGetFileAttributes_t, 545
pFixId
unpack_loc_PositionRpt_Ind_t, 449
pFlashPayLd
pack_voice_SLQSVoiceSendFlash_t, 322
pFlashType
pack_voice_SLQSVoiceSendFlash_t, 322
pFollowOnDC
sms_sendAsyncsmsParams, 366
pForceOnDC
sms_sendAsyncsmsParams, 366
pForceRev0
pack_nas_SetCDMANetworkParameters_t, 253
pFwAutoCheck
pack_swioma_SLQSOMADMSetSettings_t, 297
pFwAutoSDM
pack_swiavms_SLQSAvMSSetSettings_t, 292
pGERANInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pGPRSMinimumQoS
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pGPRSRequestedQos
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pGSMBER
unpack_nas_SLQSGetErrorRate_t, 466
pGSMCallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pGSMCipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pGSMRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pGSMRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 261
pGSMRSSIThreshList
nas_GSMRSSIThresh, 128
pGSMSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 484
pGSMSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pGSMSysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pGWAcqOrderPref
pack_nas_SLQSSetSysSelectionPref_t, 274
unpack_nas_SLQSGetSysSelectionPref_t, 478
pGetCallFWExtInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 556
pGetCallFWInfo
unpack_voice_SLQSVoiceGetCallForwarding-Status_t, 556
pGetCustomInput
DMSgetCustomFeatureV2, 52
pGetDyingGaspCfg
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 423
pGetDyingGaspStatistics
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 423
pGetIndicationRegInfo
unpack_sms_SLQSGetIndicationRegister_t, 524
pGetMsgWaitingInfoResp
unpack_sms_SLQSGetMessageWaiting_t, 524
pGetTransLayerInfo
unpack_sms_SLQSGetTransLayerInfo_t, 527
pGetTransNWRegInfo
unpack_sms_SLQSGetTransNWRegInfo_t, 527
pGnssData
pack_loc_Delete_Assist_Data_t, 238
pGpsTime
unpack_loc_BestAvailPos_Ind_t, 440
unpack_loc_PositionRpt_Ind_t, 449
pHA2002bis
pack_wds_SetMobileIPParameters_t, 334
pHAAuthenticator
pack_wds_SetMobileIPParameters_t, 334
pHASPI
pack_wds_SetMobileIPProfile_t, 335
pHDRECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRECIOThreshList
nas_HDRECIOThresh, 132
pHDRIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRIOThreshList
nas_HDRIOThresh, 133
pHDRNewUATIAssInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pHDRPackErrRate

unpack_nas_SLQSGetErrorRate_t, 466
pHDRRSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRRSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRRSSIThreshList
nas_HDRRSSIThresh, 133
pHDRSINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRSINRThresList
nas_HDRSINRThresh, 134
pHDRSINRThresh
pack_nas_SLQSConfigSigInfo_t, 256
pack_nas_SLQSNasConfigSigInfo2_t, 261
pHDRSINRThreshList
nas_HDRSINRThreshold, 135
pHDRSessionCloseInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pHDRSigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 484
pHDRSrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pHDRSysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pHTTPStatus
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
pHaltSubscription
unpack_uim_SLQSUIMGetConfiguration_t, 545
pHeading
unpack_loc_BestAvailPos_Ind_t, 440
unpack_loc_PositionRpt_Ind_t, 449
pHeadingUnc
unpack_loc_BestAvailPos_Ind_t, 440
unpack_loc_PositionRpt_Ind_t, 450
pHomeSIDNID
unpack_nas_SLQSNasGet3GPP2Subscription_t, 479
pHorCirConf
unpack_loc_BestAvailPos_Ind_t, 440
pHorConfidence
unpack_loc_PositionRpt_Ind_t, 450
pHorElpConf
unpack_loc_BestAvailPos_Ind_t, 440
pHorReliability
unpack_loc_BestAvailPos_Ind_t, 440
unpack_loc_PositionRpt_Ind_t, 450
pHorUncCircular
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pHorUncEllipseOrientAzimuth
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pHorUncEllipseSemiMajor
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pHorUncEllipseSemiMinor
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pHorizontalAccuracyLvl
pack_loc_Start_t, 251
pHotSwapStatus
unpack_uim_GetCardStatus_t, 540
pHwConfig
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 344
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 594
PI
voice_callFWExtInfo, 613
pIOThresList
nas_IOThresh, 139
pIOThresh
pack_nas_SLQSConfigSigInfo_t, 256
pIPv4AddrPref
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pIPv6AddPref
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pIgnoreHotSwapSwitch
pack_uim_SLQSUIMPowerUp_t, 304
pImCnFlag
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pImageList
pack_fms_SetImagesPreference_t, 237
unpack_fms_GetImagesPreference_t, 434
pIndicationToken
pack_uim_ChangePin_t, 298
pack_uim_ReadTransparent_t, 299
pack_uim_SetPinProtection_t, 300
pack_uim_SLQSUIMAuthenticate_t, 301
pack_uim_SLQSUIMGetFileAttributes_t, 303
pack_uim_UnblockPin_t, 308
pack_uim_VerifyPin_t, 309
unpack_uim_ChangePin_t, 539
unpack_uim_ReadTransparent_t, 541
unpack_uim_SetPinProtection_t, 542
unpack_uim_SLQSUIMAuthenticate_t, 543
unpack_uim_SLQSUIMGetFileAttributes_t, 545
unpack_uim_UnblockPin_t, 547
unpack_uim_VerifyPin_t, 548
pInfo
pack_voice_AnswerUSSD_t, 310
pack_voice_OriginateUSSD_t, 310
pIntermediateReportState
pack_loc_Start_t, 251
pInterval
unpack_pds_GetPDSDefaults_t, 497
pIpcpAckTimeout
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pIpcpCreqRetryCount
LibpackProfile3GPP2, 70

LibPackprofile_3GPP2, 81
pIsPcscfAddressNedded
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pKeyReferenceID
pack_uim_ChangePin_t, 298
pack_uim_SetPinProtection_t, 300
pack_uim_UnblockPin_t, 308
pack_uim_VerifyPin_t, 309
PLMNData
nas_operatorPLMNList, 167
PLMNName
nas_operatorNameString, 165
PLMNNetName
nas_PLMNNetworkName, 173
PLMNRecID
nas_OperatorPLMNData, 166
pLTEAttachProfile
pack_wds_SLQSSet3GPPConfigItem_t, 341
pLTEAttachProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 341
pLTEBandPref
pack_nas_SLQSSetSysSelectionPref_t, 274
unpack_nas_SLQSGetSysSelectionPref_t, 478
pLTECphyCa
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pLTEInfo
unpack_nas_SLQSNasSwiModemStatus_t, 485
pLTEInfoInterfreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pLTEInfoIntrafreq
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pLTEInfoNeighboringGSM
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pLTEInfoNeighboringWCDMA
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pLTERSRPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pLTERSRPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 261
pLTERSRPThreshList
nas_LTERSRPThresh, 149
pLTERSRQDelta
pack_nas_SLQSNasConfigSigInfo2_t, 261
pLTERSRQThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pLTERSRQThreshList
nas_LTERSRQThresh, 150
pLTERSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pLTERSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pLTERSSIThreshList
nas_LTERSSIThresh, 150
pLTESNRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pLTESNRThresList
nas_LTESNRThresh, 153
pLTESNRThresh
pack_nas_SLQSConfigSigInfo_t, 256
pack_nas_SLQSNasConfigSigInfo2_t, 262
pLTESNRThreshList
nas_LTESNRThreshold, 154
pLTESigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 484
pLTESigRptCfg
pack_nas_SLQSConfigSigInfo_t, 256
pLTESigRptConfig
pack_nas_SLQSNasConfigSigInfo2_t, 262
pLTESrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pLTESysInfo
unpack_nas_SLQSGetSysInfo_t, 474
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pLTEVoiceSupportSysInfo
unpack_nas_SLQSGetSysInfo_t, 475
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pLatitude
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pLcpAckTimeout
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pLcpCreqRetryCount
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pLeapSeconds
unpack_loc_PositionRpt_Ind_t, 450
pLinktimer
pack_sms_SendSMS_t, 281
sms_sendAsyncsmsParams, 366
pLongitude
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pLteEARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 268
unpack_nas_SLQSNASSwiGetChannelLock_t, 484
pLtePCI
pack_nas_SLQSNASSwiSetChannelLock_t, 268
unpack_nas_SLQSNASSwiGetChannelLock_t, 484
pMCC
unpack_nas_GetHomeNetwork3GPP2_t, 456
pMIPStatusInd
pack_wds_SLQSWdsSetEventReport_t, 347
pMNAAA
pack_wds_SetMobileIPProfile_t, 335
pMNC
unpack_nas_GetHomeNetwork3GPP2_t, 456
pMNCIncPCSDigStat
pack_nas_SLQSSetSysSelectionPref_t, 275
pMNHA
pack_wds_SetMobileIPProfile_t, 335
pMNRInfo

pack_nas_SLQSInitiateNetworkRegistration_t, 258
pMagneticDeviation
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pManagedRoamingInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pMaxStorageSizeReq
pack_sms_SLQSSmsGetMaxStorageSize_t, 289
pMaxStorageSizeResp
unpack_sms_SLQSSmsGetMaxStorageSize_t, 528
pMessage
pack_sms_SaveSMS_t, 281
pack_sms_SendSMS_t, 281
sms_sendAsyncsmsParams, 366
pMessageIndex
pack_sms_SLQSDeleteSMS_t, 283
unpack_sms_SaveSMS_t, 522
pMessageMode
pack_sms_SLQSDeleteSMS_t, 283
pack_sms_SLQSGetSMS_t, 284
pack_sms_SLQSGetSMSList_t, 285
pack_sms_SLQSModifySMSStatus_t, 286
sms_maxStorageSizeReq, 360
pMessageProtocol
unpack_sms_SLQSSmsGetMessageProtocol_t, 528
pMessageTag
pack_sms_SLQSDeleteSMS_t, 283
pMinBasedIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 479
pMinIntervalTime
pack_loc_Start_t, 251
pMncPcsDigitStatus
pack_nas_SLQSInitiateNetworkRegistration_t, 258
pMncPcsStatus
pack_nas_SLQSGetPLMNName_t, 257
pMode
pack_wds_SetMobileIPParameters_t, 334
pModePref
pack_nas_SLQSSetSysSelectionPref_t, 275
unpack_nas_SLQSGetSysSelectionPref_t, 478
pNAI
pack_wds_SetMobileIPProfile_t, 335
pNAMNameInfo
unpack_nas_SLQSNasGet3GPP2Subscription_t, 479
pNID
unpack_nas_GetHomeNetwork3GPP2_t, 457
pNITZInformation
unpack_nas_SLQSGetOperatorNameData_t, 468
pNamID
pack_voice_SLQSVoiceGetConfig_t, 319
pName
pack_wds_SetDefaultProfile_t, 332
unpack_nas_GetHomeNetwork3GPP2_t, 456
pNegoDnsSrvrPref
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pNeighborSetPilotPN
nas_NetworkStat1x, 162
pNetSelPref
pack_nas_SLQSSetSysSelectionPref_t, 275
unpack_nas_SLQSGetSysSelectionPref_t, 478
pNetworkRejectInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pNetworkStat1x
unpack_nas_SLQSSwiNetworkDebug_t, 490
pNetworkStatEVDO
unpack_nas_SLQSSwiNetworkDebug_t, 490
pNetworkTimeInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pNotifStore
pack_swiavms_SLQSAvMSSetSettings_t, 292
pNotification
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
_unpack_swiavms_SLQSAvMSSessionGetInfo_t, 32
pNotificationStore
unpack_swiavms_SLQSAvMSGetSettings_t, 530
pNumberOfPhySlot
unpack_uim_SLQSUIMGetSlotsStatus_t, 546
pNw2DescDisp
unpack_nas_GetHomeNetwork3GPP2_t, 457
pNw2DescEnc
unpack_nas_GetHomeNetwork3GPP2_t, 457
pNw2DescLen
unpack_nas_GetHomeNetwork3GPP2_t, 457
pNw2MCC
unpack_nas_GetHomeNetwork3GPP2_t, 457
pNw2MNC
unpack_nas_GetHomeNetwork3GPP2_t, 457
pNw2Name
unpack_nas_GetHomeNetwork3GPP2_t, 457
pOTASPStatus
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pObjectVer
unpack_nas_SLQSSwiNetworkDebug_t, 490
pOperation
unpack_pds_GetPDSDefaults_t, 497
pOperatorNameString
unpack_nas_SLQSGetOperatorNameData_t, 468
pOperatorPLMNList
unpack_nas_SLQSGetOperatorNameData_t, 468
pOptList
wds_DHCPv4OptionList, 640
wdsDhcpv4OptionList, 650
pOptVal
wds_DHCPOpt, 638
pPCSInstance
unpack_nas_PerformNetworkScan_t, 461
pPCSInstanceSize
unpack_nas_PerformNetworkScan_t, 461

pPDNInactivTimeout
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 76
pPDNInactivTimeout3GPP2
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pPDPtype
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pPLMNNetworkName
unpack_nas_SLQSGetOperatorNameData_t, 468
pPRLPref
pack_nas_SLQSSetSysSelectionPref_t, 275
unpack_nas_SLQSGetSysSelectionPref_t, 478
pPWD
_PackSwiAVMSSettingsAPNInfo, 29
pPackageID
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
_unpack_swiavms_SLQSAvMSSessionGetInfo_t, 32
pack_swiavms_SLQSAvMSSendSelection_t, 290
pPass
pack_wds_SLQSStartDataSession_t, 345
pPassword
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pack_wds_SetDefaultProfile_t, 332
pPasswordSize
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pPcscfAddrUsingDhcp
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 75
pPcscfAddrUsingPCO
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 76
pPdnType
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pPdpAccessConFlag
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 76
pPdpContext
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 76
pPdpDataCompType
LibpackProfile3GPP, 64
LibPackprofile_3GPP, 76
pPdpHdrCompType
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pPeriodInfo
pack_swiavms_SLQSAvMSSetSettings_t, 292
pPeroidsInfo
unpack_swiavms_SLQSAvMSGetSettings_t, 530
pPersonalityListLength
unpack_nas_SLQSSwiGetHDRPersonality_t, 487
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 488
pPersonalizationStatus
unpack_uim_SLQSUIMGetConfiguration_t, 545
pPilotSetData
unpack_nas_SLQSSwiGetHRPDStats_t, 489
pPilotSetInfo
nas_PilotSetData, 172
pPollingTimer
pack_swiavms_SLQSAvMSSetSettings_t, 292
unpack_swiavms_SLQSAvMSGetSettings_t, 530
pPppSessCloseTimer1x
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pPppSessCloseTimerDO
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pPrecisionDilution
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pPrefVoiceDomain
pack_voice_SLQSVoiceSetConfig_t, 324
pPrefVoicePrivacy
pack_voice_SLQSVoiceGetConfig_t, 319
pPrefVoiceSO
pack_voice_SLQSVoiceGetConfig_t, 319
pack_voice_SLQSVoiceSetConfig_t, 324
pPrefVoiceSOStatus
unpack_voice_SLQSVoiceSetConfig_t, 571
pPriDNSIPv4AddPref
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pPriDNSIPv6addpref
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pPriV6DnsAddress
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pPrimaryHA
pack_wds_SetMobileIPProfile_t, 335
pPrimaryID
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pPrimaryV4DnsAddress
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pProfileID
unpack_wds_SLQSCreateProfile_t, 583
pProfileId
pack_wds_SLQSCreateProfile_t, 336
pack_wds_SLQSModifyProfile_t, 340
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 343
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 344
pProfileList
pack_wds_SLQSSet3GPPConfigItem_t, 341
pProfileSettings
unpack_wds_SLQSGetProfileSettings_t, 588
pProfileType
pack_wds_SLQSCreateProfile_t, 336

pack_wds_SLQSModifyProfile_t, 340
pProfilename
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pProfilenameSize
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pProtoSubTypElmnt
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 488
pProtocol
pack_nas_SetCDMANetworkParameters_t, 254
pProtocolSubtypeElement
unpack_nas_SLQSSwiGetHDRPersonality_t, 487
pQosClassID
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pRAT
pack_nas_SLQSSetSysSelectionPref_t, 275
pRATINstance
unpack_nas_PerformNetworkScan_t, 461
pRATInstanceSize
unpack_nas_PerformNetworkScan_t, 461
pRATType
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
PRIString
unpack_dms_GetFirmwareRevision_t, 406
unpack_dms_GetFirmwareRevisions_t, 407
PRLInd
unpack_nas_SLQSGetServingSystem_t, 471
PRLPTlv
NASQmiCbkNasSystemSelPrefInd, 221
PRLPref
NASPRLPreferenceTlv, 220
pRSRPThresList
nas_RSRPThresh, 182
pRSRPThresh
pack_nas_SLQSConfigSigInfo_t, 256
pRSRQThresList
nas_RSRQThresh, 183
pRSRQThresh
pack_nas_SLQSConfigSigInfo_t, 256
pRSSIThresList
nas_RSSIThresh, 184
pRSSIThresh
pack_nas_SLQSConfigSigInfo_t, 256
pRXChain0Info
unpack_nas_SLQSNasGetTxRxInfo_t, 482
pRXChain1Info
unpack_nas_SLQSNasGetTxRxInfo_t, 482
pRXDroppedCount
unpack_wds_GetPacketStatistics_t, 581
pRXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 581
pRXOkBytesCount
unpack_wds_GetPacketStatistics_t, 581
pRXPacketErrors
unpack_wds_GetPacketStatistics_t, 581
pRXPacketOverflows
unpack_wds_GetPacketStatistics_t, 581
pRXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 581
pRXTotalBytes
unpack_wds_GetByteTotals_t, 574
pRadioInterface
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 483
pRankIndicatorInd
pack_nas_SLQSNasSwiIndicationRegister_t, 267
pReRegPeriod
pack_wds_SetMobileIPParameters_t, 334
pReRegTraffic
pack_wds_SetMobileIPParameters_t, 334
pReadResult
unpack_uim_ReadTransparent_t, 541
pReasonMask
unpack_dms_GetOfflineReason_t, 410
pRecurrenceType
pack_loc_Start_t, 251
pRefreshEvent
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 546
pRegCallStatInfoEvt
sms_getIndicationReg, 357
sms_setIndicationReg, 367
pRegDTMFEvents
pack_voice_SLQSVoiceIndicationRegister_t, 320
pRegInd
sms_getTransLayerInfo, 358
pRegStatus
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
sms_getTransNWRegInfo, 359
pRegTransLayerInfoEvt
sms_getIndicationReg, 357
sms_setIndicationReg, 367
pRegTransNWRegInfoEvt
sms_getIndicationReg, 357
sms_setIndicationReg, 367
pRegVoicePrivacyEvents
pack_voice_SLQSVoiceIndicationRegister_t, 320
pRejectReason
pack_swiavms_SLQSAvMSSendSelection_t, 290
pack_swioma_SLQSOMADMSendSelection_t, 296
pRemainingRetries
unpack_uim_ChangePin_t, 540
unpack_uim_SetPinProtection_t, 542
unpack_uim_SLQSUIMDepersonalization_t, 543
unpack_uim_UnblockPin_t, 547
unpack_uim_VerifyPin_t, 548
pRemotePartyName
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pRemotePartyNum
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pReportChannelRate

pack_wds_SLQSGetDUNCallInfo_t, 337
pReportConnStatus
pack_wds_SLQSGetDUNCallInfo_t, 337
pReportDataBearerTech
pack_wds_SLQSGetDUNCallInfo_t, 337
pReportDormStatus
pack_wds_SLQSGetDUNCallInfo_t, 337
pReqSettings
pack_wds_SLQSGetRuntimeSettings_t, 339
pRequestOptionList
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 344
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 595
pRequestedTag
pack_sms_SLQSGetSMSList_t, 285
pRetryInterval
pack_wds_SetMobileIPParameters_t, 334
pRetryLimit
pack_wds_SetMobileIPParameters_t, 334
pRetryMessage
sms_sendAsyncsmsParams, 366
pRetryMessageId
sms_sendAsyncsmsParams, 366
pRevInUse
nas_CDMASysInfo, 110
pRevInUseValid
nas_CDMASysInfo, 111
pRevTunneling
pack_wds_SetMobileIPProfile_t, 335
pRoamPref
pack_nas_SLQSSetSysSelectionPref_t, 275
unpack_nas_SLQSGetSysSelectionPref_t, 478
pRoamTimer
pack_voice_SLQSVoiceGetConfig_t, 319
pRoamTimerCnt
unpack_voice_SLQSVoiceGetConfig_t, 568
pRoamTimerConfig
pack_voice_SLQSVoiceSetConfig_t, 324
pRoamTimerStatus
unpack_voice_SLQSVoiceSetConfig_t, 571
pRoaming
pack_nas_SetCDMANetworkParameters_t, 254
pRscp
unpack_nas_SLQSNasSigInfoCallback_ind_t, 484
PSDomain
unpack_nas_GetServingNetwork_t, 460
pSID
unpack_nas_GetHomeNetwork3GPP2_t, 457
pSMSAttemptedFlag
packgetDyingGaspStatistics, 349
pSMSCAddress
pack_sms_SetSMSCAddress_t, 282
unpack_sms_GetSMSCAddress_t, 521
pSMSCType
pack_sms_SetSMSCAddress_t, 282
unpack_sms_GetSMSCAddress_t, 522
pSPC
pack_nas_SetCDMANetworkParameters_t, 254
pack_wds_SetMobileIPParameters_t, 334
pSV
loc_BdsSVInfo, 88
loc_SVInfo, 99
pSVInfo
pack_loc_Delete_Assist_Data_t, 238
pSarRFState
unpack_sar_SLQSGetRfSarState_t, 521
pSatelliteInfo
unpack_loc_GnssSvInfo_Ind_t, 444
pScanResult
unpack_nas_PerformNetworkScan_t, 461
pSccRxInfo
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 490
pSecDNSIPv4AddPref
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pSecDNSIPv6addpref
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pSecV6DnsAddress
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pSecondaryFlag
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pSecondaryHA
pack_wds_SetMobileIPProfile_t, 335
pSecondaryV4DnsAddress
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pSectorID
nas_NetworkStatEVDO, 164
pSendSmsParams
pack_sms_SLQSSendAsyncSMS_t, 286
pSensorDataUsage
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pServiceClass
pack_voice_SLQSVoiceSetSUPSService_t, 327
pServiceOption
sms_sendAsyncsmsParams, 366
pServingSystemInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pSessionType
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
pSetIndicationRegReq
pack_sms_SLQSSetIndicationRegister_t, 287
pSetRoutesReq
pack_sms_SLQSSmsSetRoutes_t, 289
pSigIndReq
pack_nas_SLQSSetSignalStrengthsCallback_t, 270
pSignalStrengthInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pSmsOnIms
sms_sendAsyncsmsParams, 366

pSmsStorage
unpack_sms_SLQSSwiGetSMSStorage_t, 529
pSourceIP
LibPackTFTIDParams, 84
pSpeedHorizontal
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pSpeedUnc
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pSpeedVertical
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pSpeedVerticalUnc
unpack_loc_BestAvailPos_Ind_t, 441
pSrvDomainPref
pack_nas_SLQSSetSysSelectionPref_t, 275
unpack_nas_SLQSGetSysSelectionPref_t, 478
pSrvOpt
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pSrvRegRestriction
pack_nas_SLQSSetSysSelectionPref_t, 275
pSrvcProviderName
unpack_nas_SLQSGetOperatorNameData_t, 468
pStatMask
pack_wds_GetPacketStatistics_t, 330
pSubscriptionInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pSuppressSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pSuppsNotifEvents
pack_voice_SLQSVoiceIndicationRegister_t, 320
pSvUsedforFix
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pSvcClass
pack_voice_SLQSVoiceGetCallBarring_t, 316
pack_voice_SLQSVoiceGetCallForwardingStatus-
_t, 316
pack_voice_SLQSVoiceGetCallWaiting_t, 317
unpack_voice_SLQSVoiceGetCallBarring_t, 554
unpack_voice_SLQSVoiceGetCallWaiting_t, 560
pSvcType
pack_voice_SLQSVoiceDialCall_t, 314
pSysInfoInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pSysInfoNoChange
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pSystemSelectionInd
pack_nas_SLQSNasIndicationRegisterExt_t, 266
pTDSCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMAECIOThreshList
nas_TDSCDMAECIOThresh, 196
pTDSCDMARSCPDelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMARSCPThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMARSCPThreshList
nas_TDSCDMARSCPThresh, 197
pTDSCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMARSSIThreshList
nas_TDSCDMARSSIThresh, 197
pTDSCDMASINRDelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMASINRThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pTDSCDMASINRThreshList
nas_TDSCDMASINRThresh, 198
pTDSCDMASigInfoExt
unpack_nas_SLQSNasSigInfoCallback_ind_t, 484
pTFTID1Params
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pTFTID2Params
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pTTYConfigStatus
unpack_voice_SLQSVoiceSetConfig_t, 571
pTTYMode
pack_voice_SLQSVoiceGetConfig_t, 319
pack_voice_SLQSVoiceSetConfig_t, 324
pTXDroppedCount
unpack_wds_GetPacketStatistics_t, 581
pTXInfo
unpack_nas_SLQSNasGetTxRxInfo_t, 482
pTXOKBytesLastCall
unpack_wds_GetPacketStatistics_t, 581
pTXOkBytesCount
unpack_wds_GetPacketStatistics_t, 581
pTXPacketErrors
unpack_wds_GetPacketStatistics_t, 581
pTXPacketOverflows
unpack_wds_GetPacketStatistics_t, 581
pTXPacketSuccesses
unpack_wds_GetPacketStatistics_t, 581
pTXTotalBytes
unpack_wds_GetByteTotals_t, 575
pTdscdmaBandPref
pack_nas_SLQSSetSysSelectionPref_t, 275
pTech
pack_wds_SLQSStartDataSession_t, 345
pTechnologyMask
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pTimeSrc
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pTimeStamp
packgetDyingGaspStatistics, 349
pTimeUnc

unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pTimeZone
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 483
pTimeout
unpack_pds_GetPDSDefaults_t, 497
pTimer
pack_nas_SLQSNasSwiIndicationRegister_t, 267
pTimerVal
pack_voice_SLQSVoiceSetSUPSService_t, 327
pTimestampUtc
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pTrackingStatus
unpack_pds_GetPDSState_t, 498
pTransLayerInfo
sms_getTransLayerInfo, 358
pTransferStatInd
pack_wds_SLQSGetDUNCallInfo_t, 337
pack_wds_SLQSWdsSetEventReport_t, 347
pTransferStatusReport
sms_setRoutesReq, 368
pTrueIMSI
unpack_nas_SLQSNasGet3GPP2Subscription_t, 479
pUATI
unpack_nas_SLQSSwiGetHRPDStats_t, 489
pUMTSCellID
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pUMTSInfo
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pUMTSMinQoS
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pUMTSMinQosSigInd
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pUMTSReqQoS
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pUMTSReqQoSSigInd
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pUSSDInfo
unpack_voice_SLQSOriginateUSSD_t, 548
pUUSInFo
pack_voice_SLQSVoiceDialCall_t, 314
pUUSInfo
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pUimSlotsStatus
unpack_uim_SLQSUIMGetSlotsStatus_t, 546
pUname
_PackSwiAVMSSettingsAPNInfo, 29
pUser
pack_wds_SLQSStartDataSession_t, 345
pUserData
sms_sendAsyncsmsParams, 366
pUserId
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pUserIdSize
LibpackProfile3GPP2, 70
LibPackprofile_3GPP2, 81
pUsername
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pack_wds_SetDefaultProfile_t, 332
pUsernameSize
LibpackProfile3GPP, 65
LibPackprofile_3GPP, 76
pVerboseFailReasonType
unpack_wds_SLQSStartDataSession_t, 596
pVerboseFailureReason
unpack_wds_SLQSStartDataSession_t, 596
pVertConfidence
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pVertReliability
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pVertUnc
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_PositionRpt_Ind_t, 450
pVoiceDomainPref
pack_voice_SLQSVoiceGetConfig_t, 319
pVoiceDomainPrefStatus
unpack_voice_SLQSVoiceSetConfig_t, 571
pVoicePrivacy
unpack_voice_SLQSVoiceGetAllCallInfo_t, 553
unpack_voice_SLQSVoiceGetCallInfo_t, 558
pWAMSParaChanged
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
pWCDMABER
unpack_nas_SLQSGetErrorRate_t, 466
pWCDMACallBarringSysInfo
unpack_nas_SLQSGetSysInfo_t, 475
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pWCDMACipherDomainSysInfo
unpack_nas_SLQSGetSysInfo_t, 475
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pWCDMAECIODelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pWCDMAECIOThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pWCDMAECIOThreshList
nas_WCDMAECIOThresh, 208
pWCDMAInfoLTENeighborCell
unpack_nas_SLQSNasGetCellLocationInfo_t, 480
pWCDMARSSIDelta
pack_nas_SLQSNasConfigSigInfo2_t, 262
pWCDMARSSIThresh
pack_nas_SLQSNasConfigSigInfo2_t, 262
pWCDMARSSIThreshList
nas_WCDMARSSIThresh, 209

pWCDMASigInfo
unpack_nas_SLQSNasSigInfoCallback_ind_t, 484
pWCDMASrvStatusInfo
unpack_nas_SLQSGetSysInfo_t, 475
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pWCDMASysInfo
unpack_nas_SLQSGetSysInfo_t, 475
unpack_nas_SLQSSysInfoCallback_ind_t, 493
pWcdmaUARFCN
pack_nas_SLQSNASSwiSetChannelLock_t, 268
unpack_nas_SLQSNASSwiGetChannelLock_t, 484
pXid
unpack_loc_BestAvailPos_Ind_t, 441
pack_dms_ActivateAutomatic
dms.h, 665
pack_dms_ActivateAutomatic_t, 223
actCode, 224
pack_dms_GetActivationState
dms.h, 666
pack_dms_GetBandCapability
dms.h, 666
pack_dms_GetCrashAction
dms.h, 666
pack_dms_GetCustFeature
dms.h, 667
pack_dms_GetCustFeaturesV2
dms.h, 667
pack_dms_GetCustFeaturesV2_t, 224
cust_id, 224
list_type, 224
Tlvresult, 224
pack_dms_GetDeviceCap
dms.h, 667
pack_dms_GetDeviceCapabilities
dms.h, 667
pack_dms_GetDeviceHardwareRev
dms.h, 668
pack_dms_GetDeviceMfr
dms.h, 668
pack_dms_GetDeviceSerialNumbers
dms.h, 668
pack_dms_GetFSN
dms.h, 670
pack_dms_GetFirmwareInfo
dms.h, 669
pack_dms_GetFirmwareRevision
dms.h, 669
pack_dms_GetFirmwareRevisions
dms.h, 670
pack_dms_GetHardwareRevision
dms.h, 670
pack_dms_GetIMSI
dms.h, 671
pack_dms_GetManufacturer
dms.h, 671
pack_dms_GetModelID
dms.h, 671
pack_dms_GetNetworkTime
dms.h, 672
pack_dms_GetOfflineReason
dms.h, 672
pack_dms_GetPRLVersion
dms.h, 673
pack_dms_GetPower
dms.h, 672
pack_dms_GetSerialNumbers
dms.h, 673
pack_dms_GetUSBComp
dms.h, 673
pack_dms_GetVoiceNumber
dms.h, 674
pack_dms_ResetToFactoryDefaults
dms.h, 674
pack_dms_ResetToFactoryDefaults_t, 224
spc, 225
pack_dms_SLQSDmsSwiGetResetInfo
dms.h, 677
pack_dms_SLQSDmsSwiIndicationRegister
dms.h, 678
pack_dms_SLQSDmsSwiIndicationRegister_t, 228
resetInfoInd, 228
pack_dms_SLQSGetBandCapability
dms.h, 678
pack_dms_SLQSGetERIFile
dms.h, 679
pack_dms_SLQSSwiClearDyingGaspStatistics
dms.h, 679
pack_dms_SLQSSwiGetCrashInfo
dms.h, 679
pack_dms_SLQSSwiGetCrashInfo_t, 228
clear, 229
pack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 680
pack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 680
pack_dms_SLQSSwiGetFirmwareCurr
dms.h, 680
pack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 681
pack_dms_SLQSSwiGetHostDevInfo
dms.h, 681
pack_dms_SLQSSwiGetOSInfo
dms.h, 681
pack_dms_SLQSSwiGetSerialNoExt
dms.h, 682
pack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 682
pack_dms_SLQSSwiSetDyingGaspCfg_t, 229
pDestSMSContent, 229
pDestSMSNum, 229
pack_dms_SLQSSwiSetHostDevInfo
dms.h, 682
pack_dms_SLQSSwiSetHostDevInfo_t, 229
manString, 230
modelString, 230

plasmaIDString, 230
swVerString, 230
pack_dms_SLQSSwiSetOSInfo
dms.h, 683
pack_dms_SLQSSwiSetOSInfo_t, 230
nameString, 230
versionString, 230
pack_dms_SLQSUIMGetState
dms.h, 683
pack_dms_SetActivationStatusCallback
dms.h, 675
pack_dms_SetActivationStatusCallback_t, 225
activationState, 225
pack_dms_SetCrashAction
dms.h, 675
pack_dms_SetCrashAction_t, 225
crashAction, 226
pack_dms_SetCustFeature
dms.h, 675
pack_dms_SetCustFeature_t, 226
DHCPRelayEnabled, 226
DisableIMSI, 226
GPSLPM, 226
GPSSel, 226
GpsEnable, 226
IPFamSupport, 226
IsVoiceEnabled, 226
RMAutoConnect, 226
SMSSupport, 226
pack_dms_SetCustFeaturesV2
dms.h, 676
pack_dms_SetCustFeaturesV2_t, 226
cust_id, 227
cust_value, 227
Tlvresult, 227
value_length, 227
pack_dms_SetEventReport
dms.h, 676
pack_dms_SetEventReport_t, 227
mode, 227
pack_dms_SetFirmwarePreference
dms.h, 676
pack_dms_SetPower
dms.h, 677
pack_dms_SetPower_t, 227
mode, 227
Tlvresult, 228
pack_dms_SetUSBComp
dms.h, 677
pack_dms_SetUSBComp_t, 228
Tlvresult, 228
USBComp, 228
pack_dms_UIMChangePIN
dms.h, 683
pack_dms_UIMChangePIN_t, 230
id, 231
newValue, 231
oldValue, 231
pack_dms_UIMGetControlKeyStatus
dms.h, 684
pack_dms_UIMGetControlKeyStatus_t, 231
facility, 232
pack_dms_UIMGetICCID
dms.h, 684
pack_dms_UIMGetICCID_t, 232
Tlvresult, 232
pack_dms_UIMGetPINStatus
dms.h, 685
pack_dms_UIMSetControlKeyProtection
dms.h, 685
pack_dms_UIMSetControlKeyProtection_t, 232
facility, 233
facilityCk, 233
facilityState, 233
pack_dms_UIMSetPINProtection
dms.h, 685
pack_dms_UIMSetPINProtection_t, 233
bEnable, 233
id, 233
value, 233
pack_dms_UIMUnblockControlKey
dms.h, 686
pack_dms_UIMUnblockControlKey_t, 233
facility, 234
facilityCk, 234
pack_dms_UIMUnblockPIN
dms.h, 686
pack_dms_UIMUnblockPIN_t, 234
id, 234
newPin, 235
pukValue, 235
pack_dms_UIMVerifyPIN
dms.h, 686
pack_dms_UIMVerifyPIN_t, 235
id, 235
value, 235
pack_dms_ValidateSPC
dms.h, 687
pack_fms_GetImagesPreference
fms.h, 710
pack_fms_GetImagesPreference_t, 235
Tlvresult, 235
pack_fms_GetStoredImages
fms.h, 711
pack_fms_GetStoredImages_t, 236
Tlvresult, 236
pack_fms_SetImagesPreference
fms.h, 711
pack_fms_SetImagesPreference_t, 236
bForceDownload, 237
imageListSize, 237
modemindex, 237
pImageList, 237
Tlvresult, 237
pack_loc_Delete_Assist_Data_t, 237
pBdsSVInfo, 238

pCellDb, 238
pClkInfo, 238
pGnssData, 238
pSVInfo, 238
Tlvresult, 238
pack_loc_DeleteAssistData
loc.h, 725
pack_loc_EventRegister
loc.h, 725
pack_loc_EventRegister_t, 238
eventRegister, 240
Tlvresult, 240
pack_loc_SLQSLOCGetBestAvailPos
loc.h, 726
pack_loc_SLQSLOCGetBestAvailPos_t, 241
Tlvresult, 241
xid, 241
pack_loc_SLQSLOCInjectPosition
loc.h, 726
pack_loc_SLQSLOCInjectPosition_t, 241
altitudeSrcInfo, 245
altitudeWrtEllipsoid, 245
altitudeWrtMeanSeaLevel, 245
has_altitudeSrcInfo, 245
has_altitudeWrtEllipsoid, 245
has_horConfidence, 245
has_horReliability, 245
has_horUncCircular, 245
has_latitude, 246
has_longitude, 246
has_positionSrc, 246
has_rawHorConfidence, 246
has_rawHorUncCircular, 246
has_timestampAge, 246
has_timestampUtc, 246
has_vertConfidence, 246
has_vertRelicability, 246
has_vertUnc, 246
horConfidence, 246
horReliability, 246
horUncCircular, 246
latitude, 246
longitude, 246
positionSrc, 246
rawHorConfidence, 246
rawHorUncCircular, 246
timestampAge, 246
timestampUtc, 246
vertConfidence, 246
vertReliability, 246
vertUnc, 246
pack_loc_SLQSLOCInjectSensorData
loc.h, 727
pack_loc_SLQSLOCInjectSensorData_t, 246
accelTemp, 248
acceleroData, 248
acceleroTimeSrc, 248
gyroData, 248
gyroTemp, 248
gyroTimeSrc, 248
has_accelTemp, 248
has_acceleroTimeSrc, 248
has_accleroData, 248
has_gyroData, 248
has_gyroTemp, 248
has_gyroTimeSrc, 248
has_opaqueId, 248
opaqueId, 248
pack_loc_SLQSLOCInjectUTCTime
loc.h, 727
pack_loc_SLQSLOCInjectUTCTime_t, 248
timeMsec, 249
timeUncMsec, 249
pack_loc_SLQSLOCSetCradleMountConfig
loc.h, 727
pack_loc_SLQSLOCSetCradleMountConfig_t, 249
confidence, 249
has_confidence, 249
state, 249
pack_loc_SetExtPowerState
loc.h, 725
pack_loc_SetExtPowerState_t, 240
extPowerState, 240
Tlvresult, 240
pack_loc_SetOperationMode
loc.h, 726
pack_loc_SetOperationMode_t, 240
mode, 241
Tlvresult, 241
pack_loc_Start
loc.h, 728
pack_loc_Start_t, 249
pApplicationInfo, 250
pConfigAltitudeAssumed, 251
pHorizontalAccuracyLvl, 251
pIntermediateReportState, 251
pMinIntervalTime, 251
pRecurrenceType, 251
SessionId, 251
Tlvresult, 251
pack_loc_Stop
loc.h, 728
pack_loc_Stop_t, 251
SessionId, 251
Tlvresult, 251
pack_nas_GetACCOLC
nas.h, 743
pack_nas_GetANAAAAuthenticationStatus
nas.h, 743
pack_nas_GetCDMANetworkParameters
nas.h, 744
pack_nas_GetHomeNetwork
nas.h, 744
pack_nas_GetHomeNetwork3GPP2
nas.h, 744
pack_nas_GetNetworkPreference

nas.h, 745
pack_nas_GetRFInfo
nas.h, 745
pack_nas_GetServingNetwork
nas.h, 745
pack_nas_GetServingNetworkCapabilities
nas.h, 745
pack_nas_GetSignalStrengths
nas.h, 746
pack_nas_InitiateDomainAttach
nas.h, 746
pack_nas_InitiateDomainAttach_t, 251
action, 252
pack_nas_PerformNetworkScan
nas.h, 746
pack_nas_SLQSConfigSigInfo
nas.h, 748
pack_nas_SLQSConfigSigInfo_t, 255
pECIOThresh, 256
pIOThresh, 256
pLTESNRThresh, 256
pLTESigRptCfg, 256
pRSRPThresh, 256
pRSRQThresh, 256
pRSSIThresh, 256
pack_nas_SLQSGetErrorRate
nas.h, 749
pack_nas_SLQSGetNetworkTime
nas.h, 749
pack_nas_SLQSGetOperatorNameData
nas.h, 749
pack_nas_SLQSGetPLMNName
nas.h, 750
pack_nas_SLQSGetPLMNName_t, 256
mcc, 257
mnc, 257
pMncPcsStatus, 257
pack_nas_SLQSGetServingSystem
nas.h, 750
pack_nas_SLQSGetSignalStrength
nas.h, 750
pack_nas_SLQSGetSysInfo
nas.h, 751
pack_nas_SLQSGetSysSelectionPref
nas.h, 751
pack_nas_SLQSInitiateNetworkRegistration
nas.h, 751
pack_nas_SLQSInitiateNetworkRegistration_t, 257
pChangeDuration, 257
pMNRInfo, 258
pMncPcsDigitStatus, 258
regAction, 258
pack_nas_SLQSNASSwiGetChannelLock
nas.h, 754
pack_nas_SLQSNASSwiSetChannelLock
nas.h, 755
pack_nas_SLQSNASSwiSetChannelLock_t, 267
pack_nas_SLQSNasConfigSigInfo2
nas.h, 752
pack_nas_SLQSNasConfigSigInfo2_t, 258
pHDRIODelta, 261
pHDRIOThresh, 261
pLTESigRptConfig, 262
pack_nas_SLQSNasGet3GPP2Subscription
nas.h, 752
pack_nas_SLQSNasGet3GPP2Subscription_t, 262
namID, 263
pack_nas_SLQSNasGetCellLocationInfo
nas.h, 752
pack_nas_SLQSNasGetHDRColorCode
nas.h, 753
pack_nas_SLQSNasGetSigInfo
nas.h, 753
pack_nas_SLQSNasGetTxRxInfo
nas.h, 753
pack_nas_SLQSNasGetTxRxInfo_t, 263
radio_if, 263
pack_nas_SLQSNasIndicationRegisterExt
nas.h, 754
pack_nas_SLQSNasIndicationRegisterExt_t, 263
pDDTMInd, 266
pDualStandByPrefInd, 266
pErrorRateInd, 266
pHDRSessionCloseInd, 266
pLTECphyCa, 266
pManagedRoamingInd, 266
pNetworkRejectInd, 266
pNetworkTimeInd, 266
pServingSystemInd, 266
pSignalStrengthInd, 266
pSubscriptionInfoInd, 266
pSuppressSysInfoInd, 266
pSysInfoInd, 266
pSystemSelectionInd, 266
pack_nas_SLQSNasSwiIndicationRegister
nas.h, 754
pack_nas_SLQSNasSwiIndicationRegister_t, 266
gsmUmtsDl, 267
gsmUmtsUl, 267
lteEmmDl, 267
lteEmmUl, 267
lteEsmDl, 267
lteEsmUl, 267
pRankIndicatorInd, 267
pTimer, 267
pack_nas_SLQSNasSwiModemStatus
nas.h, 755
pack_nas_SLQSSetBandPreference
nas.h, 755
pack_nas_SLQSSetBandPreference_t, 268
bandPref, 269
pack_nas_SLQSSetSignalStrengthsCallback
nas.h, 756
pack_nas_SLQSSetSignalStrengthsCallback_t, 270
bEnable, 270
pSigIndReq, 270

pack_nas_SLQSSetSysSelectionPref
nas.h, 756
pack_nas_SLQSSetSysSelectionPref_t, 270
pAcqOrderPref, 274
pBandPref, 274
pCSGID, 274
pChgDuration, 274
pEmerMode, 274
pGWAcqOrderPref, 274
pLTEBandPref, 274
pModePref, 275
pNetSelPref, 275
pPRLPref, 275
pRAT, 275
pRoamPref, 275
pSrvDomainPref, 275
pSrvRegRestriction, 275
pTdscdmaBandPref, 275
pack_nas_SLQSSwiGetHDRPersonality
nas.h, 756
pack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 757
pack_nas_SLQSSwiGetHRPDStats
nas.h, 757
pack_nas_SLQSSwiGetLteCQI
nas.h, 757
pack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 758
pack_nas_SLQSSwiNetworkDebug
nas.h, 758
pack_nas_SLQSSwiPSDetach
nas.h, 758
pack_nas_SLQSSwiPSDetach_t, 275
pDetachAction, 275
pack_nas_SetACCOLC
nas.h, 747
pack_nas_SetACCOLC_t, 252
accolc, 252
spc, 252
pack_nas_SetCDMANetworkParameters
nas.h, 747
pack_nas_SetCDMANetworkParameters_t, 252
pApplication, 253
pBroadcast, 253
pCustomSCP, 253
pForceRev0, 253
pProtocol, 254
pRoaming, 254
pSPC, 254
pack_nas_SetLURejectCallback
nas.h, 747
pack_nas_SetNetworkPreference
nas.h, 748
pack_nas_SetNetworkPreference_t, 254
Duration, 254
TechnologyPref, 254
Tlvresult, 254
pack_nas_SetRFInfoCallback
nas.h, 748
pack_nas_SlqsGetLTECphyCAInfo
nas.h, 749
pack_pds_GetPDSDefaults
pds.h, 777
pack_pds_GetPDSState
pds.h, 777
pack_pds_GetPortAutomaticTracking
pds.h, 778
pack_pds_SetPDSDefaults
pds.h, 778
pack_pds_SetPDSDefaults_t, 275
accuracy, 276
interval, 276
operation, 276
timeout, 276
pack_pds_SetPDSState
pds.h, 778
pack_pds_SetPDSState_t, 276
enable, 276
pack_pds_SetPortAutomaticTracking
pds.h, 779
pack_pds_SetPortAutomaticTracking_t, 276
bAuto, 277
pack_qmi_t, 277
msgid, 277
svc, 277
timeout, 277
xid, 277
pack_qos_SLQSQosGetNetworkStatus
qos.h, 827
pack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 827
pack_qos_SLQSQosSwiReadApnExtraParams_t, 277
apnId, 278
pack_qos_SLQSQosSwiReadDataStats
qos.h, 828
pack_qos_SLQSQosSwiReadDataStats_t, 278
apnId, 278
pack_qos_SLQSSetQosEventCallback
qos.h, 829
pack_qos_SLQSSetQosEventCallback_t, 278
enable, 279
pack_rms_GetSMSWake
rms.h, 834
pack_rms_SetSMSWake
rms.h, 834
pack_rms_SetSMSWake_t, 279
enabled, 279
wake_mask, 279
pack_sar_SLQSGetRfSarState
sar.h, 835
pack_sar_SLQSSetRfSarState
sar.h, 836
pack_sar_SLQSSetRfSarState_t, 279
RfSarState, 280
pack_sms_GetSMSCAddress
sms.h, 842

pack_sms_SLQSDeleteSMS
sms.h, 844
pack_sms_SLQSDeleteSMS_t, 282
pMessageIndex, 283
pMessageMode, 283
pMessageTag, 283
storageType, 283
pack_sms_SLQSGetIndicationRegister
sms.h, 844
pack_sms_SLQSGetMessageWaiting
sms.h, 845
pack_sms_SLQSGetSMS
sms.h, 845
pack_sms_SLQSGetSMS_t, 283
messageIndex, 284
pMessageMode, 284
storageType, 284
pack_sms_SLQSGetSMSList
sms.h, 846
pack_sms_SLQSGetSMSList_t, 284
pMessageMode, 285
pRequestedTag, 285
storageType, 285
pack_sms_SLQSGetSmsBroadcastConfig
sms.h, 845
pack_sms_SLQSGetSmsBroadcastConfig_t, 284
mode, 284
pack_sms_SLQSGetTransLayerInfo
sms.h, 846
pack_sms_SLQSGetTransNWRegInfo
sms.h, 847
pack_sms_SLQSModifySMSStatus
sms.h, 847
pack_sms_SLQSModifySMSStatus_t, 285
messageIndex, 286
messageTag, 286
pMessageMode, 286
storageType, 286
pack_sms_SLQSSendAsyncSMS
sms.h, 847
pack_sms_SLQSSendAsyncSMS_t, 286
pSendSmsParams, 286
pack_sms_SLQSSetIndicationRegister
sms.h, 848
pack_sms_SLQSSetIndicationRegister_t, 286
pSetIndicationRegReq, 287
pack_sms_SLQSSetSmsBroadcastActivation
sms.h, 848
pack_sms_SLQSSetSmsBroadcastActivation_t, 287
broadcastActivate, 287
mode, 287
pack_sms_SLQSSetSmsBroadcastConfig
sms.h, 848
pack_sms_SLQSSetSmsBroadcastConfig_t, 287
mode, 288
pBroadcastConfig, 288
pCDMABroadcastConfig, 288
pack_sms_SLQSSetSmsStorage
sms.h, 849
pack_sms_SLQSSetSmsStorage_t, 288
smsStorage, 288
pack_sms_SLQSSmsGetMaxStorageSize
sms.h, 849
pack_sms_SLQSSmsGetMaxStorageSize_t, 288
pMaxStorageSizeReq, 289
pack_sms_SLQSSmsGetMessageProtocol
sms.h, 849
pack_sms_SLQSSmsSetRoutes
sms.h, 850
pack_sms_SLQSSmsSetRoutes_t, 289
pSetRoutesReq, 289
pack_sms_SLQSSwiGetSMSStorage
sms.h, 850
pack_sms_SaveSMS
sms.h, 842
pack_sms_SaveSMS_t, 280
messageFormat, 281
messageSize, 281
pMessage, 281
storageType, 281
pack_sms_SendSMS
sms.h, 843
pack_sms_SendSMS_t, 281
messageFormat, 281
messageSize, 281
pLinktimer, 281
pMessage, 281
pack_sms_SetNewSMSCallback
sms.h, 843
pack_sms_SetNewSMSCallback_t, 281
status, 282
pack_sms_SetSMSCAddress
sms.h, 844
pack_sms_SetSMSCAddress_t, 282
pSMSCAddress, 282
pSMSCType, 282
pack_swiavms_SLQSAvMSGetSettings
swiavms.h, 868
pack_swiavms_SLQSAvMSSendSelection
swiavms.h, 869
pack_swiavms_SLQSAvMSSendSelection_t, 289
pClientPerformOperationFlag, 290
pDeferTime, 290
pPackageID, 290
pRejectReason, 290
selection, 290
pack_swiavms_SLQSAvMSSessionGetInfo
swiavms.h, 869
pack_swiavms_SLQSAvMSSetSettings
swiavms.h, 870
pack_swiavms_SLQSAvMSSetSettings_t, 290
AutoConnect, 291
AutoReboot, 291
FwAutoUpdate, 291
FwAutodownload, 291
pAPNInfo, 291

pConnectionRetryTimers, 291
pFwAutoSDM, 292
pNotifStore, 292
pPeriodInfo, 292
pPollingTimer, 292
pack_swiavms_SLQSAvMSSetSettingsNoAutoReboot-Field
swiavms.h, 871
pack_swiavms_SLQSAvMSStartSession
swiavms.h, 872
pack_swiavms_SLQSAvMSStartSession_t, 292
sessionType, 292
pack_swiavms_SLQSAvMSStopSession
swiavms.h, 872
pack_swiavms_SLQSAvMSStopSession_t, 292
sessionType, 293
pack_swiavms_SLQSAvmsSetEventReport
swiavms.h, 870
pack_swiloc_SwiLocGetAutoStart
swiloc.h, 878
pack_swiloc_SwiLocSetAutoStart
swiloc.h, 879
pack_swiloc_SwiLocSetAutoStart_t, 293
fix_rate, 294
fix_type, 294
function, 294
max_dist, 294
max_time, 294
set_fix_rate, 294
set_fix_type, 294
set_function, 294
set_max_dist, 294
set_max_time, 294
pack_swioma_SLQSOMADMAlertCallback
swioma.h, 881
pack_swioma_SLQSOMADMCancelSession
swioma.h, 881
pack_swioma_SLQSOMADMCancelSession_t, 294
sessionType, 295
pack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 882
pack_swioma_SLQSOMADMGetSessionInfo_t, 295
SessionType, 295
pack_swioma_SLQSOMADMGetSettings
swioma.h, 883
pack_swioma_SLQSOMADMSendSelection
swioma.h, 883
pack_swioma_SLQSOMADMSendSelection_t, 295
pDeferTime, 296
pRejectReason, 296
selection, 296
pack_swioma_SLQSOMADMSetSettings
swioma.h, 884
pack_swioma_SLQSOMADMSetSettings_t, 296
FOTAdownload, 297
pAutosdm, 297
pFwAutoCheck, 297
pack_swioma_SLQSOMADMStartSession
swioma.h, 884
pack_swioma_SLQSOMADMStartSession_t, 297
sessionType, 297
pack_uim_ChangePin
uim.h, 892
pack_uim_ChangePin_t, 297
changePIN, 298
EncryptedPIN1, 298
pIndicationToken, 298
pKeyReferenceID, 298
sessionInfo, 298
Tlvresult, 298
pack_uim_GetCardStatus
uim.h, 892
pack_uim_ReadTransparent
uim.h, 893
pack_uim_ReadTransparent_t, 298
fileIndex, 299
pEncryptData, 299
pIndicationToken, 299
readTransparent, 299
sessionInfo, 299
Tlvresult, 299
pack_uim_SLQSUIMAuthenticate
uim.h, 893
pack_uim_SLQSUIMAuthenticate_t, 301
authData, 301
pIndicationToken, 301
sessionInfo, 301
pack_uim_SLQSUIMDepersonalization
uim.h, 894
pack_uim_SLQSUIMDepersonalization_t, 301
depersonilisationInfo, 302
pack_uim_SLQSUIMEventRegister
uim.h, 894
pack_uim_SLQSUIMEventRegister_t, 302
eventMask, 302
pack_uim_SLQSUIMGetConfiguration
uim.h, 894
pack_uim_SLQSUIMGetConfiguration_t, 302
pConfigurationMask, 302
pack_uim_SLQSUIMGetFileAttributes
uim.h, 895
pack_uim_SLQSUIMGetFileAttributes_t, 303
fileIndex, 303
pIndicationToken, 303
sessionInfo, 303
pack_uim_SLQSUIMGetSlotsStatus
uim.h, 895
pack_uim_SLQSUIMPowerDown
uim.h, 896
pack_uim_SLQSUIMPowerDown_t, 303
slot, 304
pack_uim_SLQSUIMPowerUp
uim.h, 896
pack_uim_SLQSUIMPowerUp_t, 304
pIgnoreHotSwapSwitch, 304
slot, 304

pack_uim_SLQSUIMRefreshComplete
uim.h, 896
pack_uim_SLQSUIMRefreshComplete_t, 304
refreshComplete, 305
sessionInfo, 305
pack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 897
pack_uim_SLQSUIMRefreshGetLastEvent_t, 305
sessionInfo, 305
pack_uim_SLQSUIMRefreshOK
uim.h, 897
pack_uim_SLQSUIMRefreshOK_t, 305
OKtoRefresh, 306
sessionInfo, 306
pack_uim_SLQSUIMRefreshRegister
uim.h, 897
pack_uim_SLQSUIMRefreshRegister_t, 306
regRefresh, 306
sessionInfo, 306
pack_uim_SLQSUIMReset
uim.h, 898
pack_uim_SLQSUIMSwitchSlot
uim.h, 898
pack_uim_SLQSUIMSwitchSlot_t, 306
bLogicalSlot, 307
ulPhysicalSlot, 307
pack_uim_SetPinProtection
uim.h, 893
pack_uim_SetPinProtection_t, 300
EncryptedPIN1, 300
pIndicationToken, 300
pKeyReferenceID, 300
pinProtection, 300
sessionInfo, 300
Tlvresult, 300
pack_uim_UnblockPin
uim.h, 898
pack_uim_UnblockPin_t, 307
EncryptedPIN1, 308
pIndicationToken, 308
pKeyReferenceID, 308
pinProtection, 308
sessionInfo, 308
Tlvresult, 308
pack_uim_VerifyPin
uim.h, 899
pack_uim_VerifyPin_t, 308
pEncryptedPIN1, 309
pIndicationToken, 309
pKeyReferenceID, 309
sessionInfo, 309
Tlvresult, 309
verifyPIN, 309
pack_voice_AnswerUSSD
voice.h, 911
pack_voice_AnswerUSSD_t, 309
pInfo, 310
pack_voice_CancelUSSD
voice.h, 911
pack_voice_OriginateUSSD
voice.h, 912
pack_voice_OriginateUSSD_t, 310
pInfo, 310
pack_voice_SLQSOriginateUSSD
voice.h, 912
pack_voice_SLQSOriginateUSSD_t, 310
ussDCS, 311
ussData, 310
ussLen, 311
pack_voice_SLQSVoiceALSSelectLine
voice.h, 913
pack_voice_SLQSVoiceALSSelectLine_t, 311
lineValue, 311
pack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 913
pack_voice_SLQSVoiceALSSetLineSwitching_t, 311
switchOption, 311
pack_voice_SLQSVoiceAnswerCall
voice.h, 913
pack_voice_SLQSVoiceAnswerCall_t, 311
pCallId, 312
pack_voice_SLQSVoiceBindSubscription
voice.h, 914
pack_voice_SLQSVoiceBindSubscription_t, 312
subsType, 312
pack_voice_SLQSVoiceBurstDTMF
voice.h, 914
pack_voice_SLQSVoiceBurstDTMF_t, 312
pack_voice_SLQSVoiceDialCall
voice.h, 914
pack_voice_SLQSVoiceDialCall_t, 313
callNumber, 314
pCLIRType, 314
pCUGInfo, 314
pCallPartySubAdd, 314
pCallType, 314
pEmergencyCategory, 314
pSvcType, 314
pUUSInFo, 314
pack_voice_SLQSVoiceEndCall
voice.h, 915
pack_voice_SLQSVoiceEndCall_t, 315
pCallId, 315
pack_voice_SLQSVoiceGetAllCallInfo
voice.h, 915
pack_voice_SLQSVoiceGetCLIP
voice.h, 917
pack_voice_SLQSVoiceGetCLIR
voice.h, 917
pack_voice_SLQSVoiceGetCNAP
voice.h, 918
pack_voice_SLQSVoiceGetCOLP
voice.h, 918
pack_voice_SLQSVoiceGetCOLR
voice.h, 918
pack_voice_SLQSVoiceGetCallBarring

voice.h, 915
pack_voice_SLQSVoiceGetCallBarring_t, 315
pSvcClass, 316
reason, 316
pack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 916
pack_voice_SLQSVoiceGetCallForwardingStatus_t, 316
pSvcClass, 316
Reason, 316
pack_voice_SLQSVoiceGetCallInfo
voice.h, 916
pack_voice_SLQSVoiceGetCallInfo_t, 316
callID, 317
pack_voice_SLQSVoiceGetCallWaiting
voice.h, 917
pack_voice_SLQSVoiceGetCallWaiting_t, 317
pSvcClass, 317
pack_voice_SLQSVoiceGetConfig
voice.h, 919
pack_voice_SLQSVoiceGetConfig_t, 317
pAMRStatus, 319
pAirTimer, 319
pAutoAnswer, 319
pNamID, 319
pPrefVoicePrivacy, 319
pPrefVoiceSO, 319
pRoamTimer, 319
pTTYMode, 319
pVoiceDomainPref, 319
pack_voice_SLQSVoiceIndicationRegister
voice.h, 919
pack_voice_SLQSVoiceIndicationRegister_t, 319
pRegDTMFEvents, 320
pRegVoicePrivacyEvents, 320
pSuppsNotifEvents, 320
pack_voice_SLQSVoiceManageCalls
voice.h, 919
pack_voice_SLQSVoiceManageCalls_t, 320
pCallID, 320
SUPSType, 321
pack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 920
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 321
pack_voice_SLQSVoiceSendFlash
voice.h, 920
pack_voice_SLQSVoiceSendFlash_t, 321
pCallID, 322
pFlashPayLd, 322
pFlashType, 322
pack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 920
pack_voice_SLQSVoiceSetCallBarringPassword_t, 322
newPasswd, 323
newPasswdAgain, 323
oldPasswd, 323
Reason, 323
pack_voice_SLQSVoiceSetConfig
voice.h, 921
pack_voice_SLQSVoiceSetConfig_t, 323
pAirTimerConfig, 324
pAutoAnswer, 324
pPrefVoiceDomain, 324
pPrefVoiceSO, 324
pRoamTimerConfig, 324
pTTYMode, 324
pack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 921
pack_voice_SLQSVoiceSetPreferredPrivacy_t, 324
privacyPref, 324
pack_voice_SLQSVoiceSetSUPSService
voice.h, 922
pack_voice_SLQSVoiceSetSUPSService_t, 324
pCallBarringPasswd, 327
pCallForwardingNumber, 327
pServiceClass, 327
pTimerVal, 327
reason, 327
voiceSvc, 327
pack_voice_SLQSVoiceStartContDTMF
voice.h, 922
pack_voice_SLQSVoiceStartContDTMF_t, 327
DTMFdigit, 327
pCallID, 327
pack_voice_SLQSVoiceStopContDTMF
voice.h, 922
pack_voice_SLQSVoiceStopContDTMF_t, 328
callID, 328
pack_wds_DHCPv4ClientLeaseChange
wds.h, 940
pack_wds_DHCPv4ClientLeaseChange_t, 328
pEnableNotification, 328
pack_wds_GetAutoconnect
wds.h, 940
pack_wds_GetByteTotals
wds.h, 940
pack_wds_GetConnectionRate
wds.h, 941
pack_wds_GetDataBearerTechnology
wds.h, 941
pack_wds_GetDefaultProfile
wds.h, 942
pack_wds_GetDefaultProfile_t, 328
profiletype, 329
pack_wds_GetDefaultProfileNum
wds.h, 942
pack_wds_GetDefaultProfileNum_t, 329
family, 329
type, 329
pack_wds_GetDormancyState
wds.h, 942
pack_wds_GetDormancyState_t, 329
pack_wds_GetLastMobileIPError
wds.h, 943
pack_wds_GetLastMobileIPError_t, 329
pack_wds_GetMobileIP
wds.h, 943

pack_wds_GetMobileIP_t, 329
pack_wds_GetMobileIPProfile
wds.h, 944
pack_wds_GetMobileIPProfile_t, 329
index, 330
pack_wds_GetPacketStatistics
wds.h, 944
pack_wds_GetPacketStatistics_t, 330
pStatMask, 330
pack_wds_GetPacketStatus
wds.h, 944
pack_wds_GetPacketStatus_t, 330
statmask, 330
pack_wds_GetSessionDuration
wds.h, 945
pack_wds_GetSessionDuration_t, 330
pack_wds_GetSessionState
wds.h, 945
pack_wds_RMSetTransferStatistics
wds.h, 946
pack_wds_RMSetTransferStatistics_t, 330
RmTrasnferStaticsReq, 331
pack_wds_SLQSCreateProfile
wds.h, 949
pack_wds_SLQSCreateProfile_t, 335
pCurProfile, 336
pProfileId, 336
pProfileType, 336
pack_wds_SLQSDeleteProfile
wds.h, 949
pack_wds_SLQSDeleteProfile_t, 336
profileIndex, 336
profileType, 336
pack_wds_SLQSGet3GPPConfigItem
wds.h, 949
pack_wds_SLQSGetCurrDataSystemStat
wds.h, 950
pack_wds_SLQSGetCurrDataSystemStat_t, 336
pack_wds_SLQSGetCurrentChannelRate
wds.h, 950
pack_wds_SLQSGetDUNCallInfo
wds.h, 951
pack_wds_SLQSGetDUNCallInfo_t, 337
Mask, 337
pReportChannelRate, 337
pReportConnStatus, 337
pReportDataBearerTech, 337
pReportDormStatus, 337
pTransferStatInd, 337
pack_wds_SLQSGetDataBearerTechnology
wds.h, 951
pack_wds_SLQSGetDataBearerTechnology_t, 337
pack_wds_SLQSGetProfileSettings
wds.h, 952
pack_wds_SLQSGetProfileSettings_t, 337
ProfileId, 338
ProfileType, 338
pack_wds_SLQSGetRuntimeSettings
wds.h, 952
pack_wds_SLQSGetRuntimeSettings_t, 338
pReqSettings, 339
pack_wds_SLQSModifyProfile
wds.h, 952
pack_wds_SLQSModifyProfile_t, 339
curProfile, 340
pProfileId, 340
pProfileType, 340
pack_wds_SLQSResetPacketStatics
wds.h, 953
pack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 954
pack_wds_SLQSSGetDHCPv4ClientConfig_t, 343
pProfileId, 343
pack_wds_SLQSSGetLoopback
wds.h, 955
pack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 955
pack_wds_SLQSSSetDHCPv4ClientConfig_t, 343
pHwConfig, 344
pProfileId, 344
pack_wds_SLQSSSetLoopback
wds.h, 956
pack_wds_SLQSSSetLoopback_t, 344
loopbackMode, 344
loopbackMultiplier, 344
pack_wds_SLQSSet3GPPConfigItem
wds.h, 953
pack_wds_SLQSSet3GPPConfigItem_t, 340
p3gppRelease, 341
pProfileList, 341
pack_wds_SLQSSetIPFamilyPreference
wds.h, 954
pack_wds_SLQSSetIPFamilyPreference_t, 342
IPFamilyPreference, 342
pack_wds_SLQSSetWdsEventCallback
wds.h, 954
pack_wds_SLQSSetWdsEventCallback_t, 342
currentDataBearer, 342
dataBearer, 342
dataBearerTechExt, 343
dataSystemStatus, 343
dormancyStatus, 343
interval, 343
mobileIP, 343
transferStats, 343
pack_wds_SLQSStartDataSession
wds.h, 956
pack_wds_SLQSStartDataSession_t, 344
pAuth, 345
pPass, 345
pTech, 345
pUser, 345
pprofileid3gpp, 345
pprofileid3gpp2, 345
pack_wds_SLQSStopDataSession
wds.h, 956

pack_wds_SLQSStopDataSession_t, 345
psid, 346
pack_wds_SLQSWdsGoActive
wds.h, 957
pack_wds_SLQSWdsGoDormant
wds.h, 957
pack_wds_SLQSWdsSetEventReport
wds.h, 958
pack_wds_SLQSWdsSetEventReport_t, 346
pCurrChannelRateInd, 347
pCurrDataBearerTechInd, 347
pCurrPrefDataSysInd, 347
pDataBearerTechInd, 347
pDataCallStatusChangeInd, 347
pDataSystemStatusChangeInd, 347
pDormancyStatusInd, 347
pMIPStatusInd, 347
pTransferStatInd, 347
pack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 958
pack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 348
contextId, 348
contextType, 348
pack_wds_SetAutoconnect
wds.h, 946
pack_wds_SetAutoconnect_t, 331
acroamsetting, 331
acsetting, 331
pack_wds_SetDefaultProfile
wds.h, 946
pack_wds_SetDefaultProfile_t, 331
authentication, 332
ipAddress, 332
pApnname, 332
pName, 332
pPassword, 332
pUsername, 332
pdpType, 332
primaryDNS, 332
profileType, 332
secondaryDNS, 332
pack_wds_SetDefaultProfileNum
wds.h, 947
pack_wds_SetDefaultProfileNum_t, 332
family, 332
index, 332
type, 332
pack_wds_SetMobileIP
wds.h, 947
pack_wds_SetMobileIP_t, 332
mode, 333
pack_wds_SetMobileIPParameters
wds.h, 948
pack_wds_SetMobileIPParameters_t, 333
pHA2002bis, 334
pHAAuthenticator, 334
pMode, 334
pReRegPeriod, 334
pReRegTraffic, 334
pRetryInterval, 334
pRetryLimit, 334
pSPC, 334
pack_wds_SetMobileIPProfile
wds.h, 948
pack_wds_SetMobileIPProfile_t, 334
index, 335
pAAASPI, 335
pAddress, 335
pEnabled, 335
pHASPI, 335
pMNAAA, 335
pMNHA, 335
pNAI, 335
pPrimaryHA, 335
pRevTunneling, 335
pSecondaryHA, 335
spc, 335
PackCreateProfileOut, 348
ExtErrorCode, 348
ProfileIndex, 348
ProfileType, 348
PackSwiAVMSSettingsAPNInfo
swiavms.h, 862
PackSwiAVMSSettingsConnectionRetryTimers
swiavms.h, 862
PackSwiAVMSSettingsPeriodsInfo
swiavms.h, 862
PackSwiAvmsSetSettingsAPNInfo
swiavms.h, 862
PackSwiAvmsSetSettingsConnectionRetryTimers
swiavms.h, 862
PackSwiAvmsSetSettingsPeriodInfo
swiavms.h, 862
package_name
unpack_omaDmFotaTlv_t, 496
packageid_str
unpack_dms_GetFirmwareInfo_t, 406
packetZone
nas_CDMASysInfo, 110
packetZoneValid
nas_CDMASysInfo, 110
packgetDyingGaspCfg, 348
pDestSMSContent, 349
pDestSMSNum, 349
packgetDyingGaspStatistics, 349
pSMSAttemptedFlag, 349
pTimeStamp, 349
path
uim_fileInfo, 389
pathLen
uim_fileInfo, 389
pbAuto
unpack_pds_GetPortAutomaticTracking_t, 498
pbPlatform
unpack_dms_GetOfflineReason_t, 410
pci

nas_cellParams, 112
nas_ltePCI, 148
nas_PhyCaAggPcellInfo, 168
nas_PhyCaAggScellIndType, 169
nas_PhyCaAggScellInfo, 172
nas_umtsLTENbrCell, 205
NASPhyCaAggPcellInfo, 217
NASPhyCaAggScellIndType, 218
NASPhyCaAggScellInfo, 219
pcsfFQDNAddress
wds_PCSCFFQDNAddressList, 644
pdpType
pack_wds_SetDefaultProfile_t, 332
pdptype
unpack_wds_GetDefaultProfile_t, 577
pds.h, 776
pack_pds_GetPDSDefaults, 777
pack_pds_GetPDSState, 777
pack_pds_GetPortAutomaticTracking, 778
pack_pds_SetPDSDefaults, 778
pack_pds_SetPDSState, 778
pack_pds_SetPortAutomaticTracking, 779
unpack_pds_GetPDSDefaults, 779
unpack_pds_GetPDSState, 779
unpack_pds_GetPortAutomaticTracking, 780
unpack_pds_SetPDSDefaults, 780
unpack_pds_SetPDSState, 780
unpack_pds_SetPortAutomaticTracking, 781
peakRate
unpack_qos_tokenBucket_t, 517
peakThroughputClass
LibPackGPRSRequestedQoS, 59
wds_GPRSQoS, 642
PersistentTechPref
unpack_nas_GetNetworkPreference_t, 458
persoFeature
appStats, 44
uim_appStatus, 378
persoRetries
appStats, 44
uim_appStatus, 378
persoState
appStats, 44
uim_appStatus, 378
persoUnblockRetries
appStats, 44
uim_appStatus, 378
pfailureCause
unpack_voice_SLQSOriginateUSSD_t, 548
phase
nas_rxInfo, 185
PhyCaAggPcellInfo
NasGetLTECphyCaInfo, 214
PhyCaAggScellDlBw
NasGetLTECphyCaInfo, 214
PhyCaAggScellIndType
NasGetLTECphyCaInfo, 214
PhyCaAggScellIndex
NasGetLTECphyCaInfo, 214
PhyCaAggScellInfo
NasGetLTECphyCaInfo, 214
PhysicalLayer
nas_protocolSubtypeElement, 177
PilotEnergy
nas_NetworkStatEVDO, 164
PilotPN
nas_PilotSetParams, 173
PilotStrength
nas_PilotSetParams, 173
PilotType
nas_PilotSetParams, 173
pin1Len
uim_encryptedPIN1, 385
pin1Retries
appStats, 44
uim_appStatus, 379
pin1State
appStats, 44
uim_appStatus, 379
pin1Val
uim_encryptedPIN1, 385
pin2Retries
appStats, 44
uim_appStatus, 379
pin2State
appStats, 44
uim_appStatus, 379
pinID
uim_changeUIMPIN, 383
uim_setPINProtection, 396
uim_unblockUIMPIN, 399
uim_verifyUIMPIN, 400
pinLen
uim_changeUIMPIN, 383
uim_verifyUIMPIN, 400
pinLength
uim_setPINProtection, 396
pinOperation
uim_setPINProtection, 396
pinProtection
pack_uim_SetPinProtection_t, 300
pack_uim_UnblockPin_t, 308
pinVal
uim_changeUIMPIN, 383
uim_verifyUIMPIN, 400
pinValue
uim_setPINProtection, 396
PkgDescLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 536
PkgDescription
unpack_swioma_SLQSOMADMGetSessionInfo_t, 536
PkgName
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537

PkgNameLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
pkgver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 424
PktErrRate
unpack_qos_swiQosFlow_t, 517
plasmaIDString
pack_dms_SLQSSwiSetHostDevInfo_t, 230
unpack_dms_SLQSSwiGetHostDevInfo_t, 426
plmn
nas_GERANInfo, 126
nas_LTEInfoIntrafreq, 146
nas_UMTSInfo, 203
port
unpack_qos_Port_t, 502
positionSrc
pack_loc_SLQSLOCInjectPosition_t, 246
pprofileid3gpp
pack_wds_SLQSStartDataSession_t, 345
pprofileid3gpp2
pack_wds_SLQSStartDataSession_t, 345
prDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
prDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
prPCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
prPCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
Precedence
unpack_qos_swiQosFilter_t, 513
precedenceClass
LibPackGPRSRequestedQoS, 59
wds_GPRSQoS, 642
prefNetwork
unpack_wds_SLQSGetCurrDataSystemStat_t, 586
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
Preferred
nas_QmiNas3GppNetworkInfo, 178
prefixLen
unpack_qos_IPv6Addr_t, 500
presentationInd
voice_remotePartyNum, 632
priChA
nas_CDMAChannel, 105
priChB
nas_CDMAChannel, 105
priSize
unpack_dms_GetFirmwareRevisions_t, 407
pridns
unpack_wds_GetDefaultProfile_t, 577
pridnsv6
unpack_wds_GetDefaultProfile_t, 577
primaryDNS
pack_wds_SetDefaultProfile_t, 332
PrimaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 590
PrimaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 590
primaryHA
unpack_wds_GetMobileIPProfile_t, 579
privacyPref
pack_voice_SLQSVoiceSetPreferredPrivacy_t, 324
priver
unpack_dms_SLQSSwiGetFirmwareCurr_t, 424
priversion_str
unpack_dms_GetFirmwareInfo_t, 406
ProfileID
unpack_wds_SLQSGetRuntimeSettings_t, 590
ProfileId
pack_wds_SLQSGetProfileSettings_t, 338
profileId
wds_DHCPProfileIdTlv, 638
wds_DHCPv4ProfileId, 640
wdsDhcpv4ProfileId, 651
ProfileId3GPP2
unpack_qos_swiQosFlow_t, 517
ProfileIdTlv
unpack_wds_DHCPv4ClientLease_ind_t, 574
ProfileIndex
PackCreateProfileOut, 348
profileIndex
pack_wds_SLQSDeleteProfile_t, 336
wds_ProfileIdentifier, 645
profileList
unpack_wds_SLQSGet3GPPConfigItem_t, 585
ProfileName
unpack_wds_SLQSGetRuntimeSettings_t, 590
ProfileType
pack_wds_SLQSGetProfileSettings_t, 338
PackCreateProfileOut, 348
unpack_wds_SLQSGetProfileSettings_t, 589
profileType
pack_wds_SetDefaultProfile_t, 332
pack_wds_SLQSDeleteProfile_t, 336
wds_DHCPProfileIdTlv, 638
wds_DHCPv4ProfileId, 640
wds_ProfileIdentifier, 645
wdsDhcpv4ProfileId, 651
profiletype
pack_wds_GetDefaultProfile_t, 329
Protocol
unpack_nas_GetCDMANetworkParameters_t, 455
ProvisionStatus
voice_CLIPResp, 617
voice_CLIRResp, 617
voice_CNAPResp, 618
voice_COLPResp, 619

voice_COLRResp, 619
psAttachState
nas_servSystem, 190
NASServingSystemInfo, 223
psBarStatus
nas_CallBarringSysInfo, 103
nas_callBarStatus, 104
psState
nas_CommInfo, 114
psc
nas_UMTSInfo, 203
nas_wcdmaCellInfo, 207
nas_WCDMASysInfo, 212
pscValid
nas_WCDMASysInfo, 212
pscsfIPv4Addr
wds_PCSCFIPv4ServerAddressList, 645
psetting
unpack_wds_GetAutoconnect_t, 574
psid
pack_wds_SLQSStopDataSession_t, 346
unpack_wds_SLQSStartDataSession_t, 596
puk1Retries
appStats, 44
uim_appStatus, 379
puk2Retries
appStats, 44
uim_appStatus, 379
pukLen
uim_unblockUIMPIN, 399
pukVal
uim_unblockUIMPIN, 400
pukValue
pack_dms_UIMUnblockPIN_t, 235

QCI
LibPackQosClassID, 82
QFlowState
unpack_qos_QosFlowInfo_t, 503
QLIC
nas_DeviceConfigDetail, 121
QMI pack/unpack (lite-qmi), 23
qaGobiApiTableBandClasses.h, 781
qaGobiApiTableCallControlReturnReasons.h, 784
qaGobiApiTableCallEndReasons.h, 785
qaGobiApiTableCarrierCodes.h, 800
qaGobiApiTableCodingScheme.h, 802
qaGobiApiTableGpsCapabilityCodes.h, 805
qaGobiApiTablePowerModes.h, 805
qaGobiApiTableRadioInterfaces.h, 806
qaGobiApiTableRegionCodes.h, 807
qaGobiApiTableServiceOptions.h, 807
qaGobiApiTableSupServiceInfoClasses.h, 809
qaGobiApiTableSwiAudio.h, 810
qaGobiApiTableSwiOMADMUpdateCompleteStatus.h, 810
qaGobiApiTableVoiceCallEndReasons.h, 812
qm_wds_ds_profile_extended_err_codes
qmerrno.h, 824
qmerrno.h
eQCWWAN_ERR_API_MUTEX_TIMEOUT, 821
eQCWWAN_ERR_BUFFER_SZ, 820
eQCWWAN_ERR_CANCEL_OP, 821
eQCWWAN_ERR_DRIVER, 821
eQCWWAN_ERR_ENUM_BEGIN, 820
eQCWWAN_ERR_ENUM_END, 821
eQCWWAN_ERR_FILE_COPY, 820
eQCWWAN_ERR_FILE_OPEN, 820
eQCWWAN_ERR_GENERAL, 820
eQCWWAN_ERR_INTERNAL, 820
eQCWWAN_ERR_INVALID_ARG, 820
eQCWWAN_ERR_INVALID_DEVID, 820
eQCWWAN_ERR_INVALID_FILE, 820
eQCWWAN_ERR_INVALID_QMI_RSP, 820
eQCWWAN_ERR_INVALID_XID, 821
eQCWWAN_ERR_MALFORMED_QMI_RSP, 820
eQCWWAN_ERR_MEMORY, 820
eQCWWAN_ERR_MULTIPLE_DEVICES, 821
eQCWWAN_ERR_MULTIPLE_SMS_UNSUPPORTED, 821
eQCWWAN_ERR_NO_CANCELABLE_OP, 821
eQCWWAN_ERR_NO_CONNECTION, 820
eQCWWAN_ERR_NO_DEVICE, 820
eQCWWAN_ERR_NO_SIGNAL, 821
eQCWWAN_ERR_NONE, 820
eQCWWAN_ERR_NULL_TLV, 824
eQCWWAN_ERR_OFFLINE, 821
eQCWWAN_ERR_PDU_GENERATION, 821
eQCWWAN_ERR_QMI_ABORTED, 821
eQCWWAN_ERR_QMI_ACCESS_DENIED, 823
eQCWWAN_ERR_QMI_ACK_NOT_SENT, 823
eQCWWAN_ERR_QMI_ARG_TOO_LONG, 821
eQCWWAN_ERR_QMI_AUTHENTICATION_FAILED, 822
eQCWWAN_ERR_QMI_AUTHENTICATION_LOCK, 822
eQCWWAN_ERR_QMI_BUNDLING_NOT_SUPPORTED, 823
eQCWWAN_ERR_QMI_CALL_FAILED, 821
eQCWWAN_ERR_QMI_CARD_BUSY_RSP, 824
eQCWWAN_ERR_QMI_CARD_CALL_CONTROL_FAILED, 823
eQCWWAN_ERR_QMI_CAT_END, 824
eQCWWAN_ERR_QMI_CAT_START, 824
eQCWWAN_ERR_QMI_CAUSE_CODE, 822
eQCWWAN_ERR_QMI_CLIENT_IDS_EXHAUSTED, 821
eQCWWAN_ERR_QMI_CONNECT, 820
eQCWWAN_ERR_QMI_DEVICE_IN_USE, 821
eQCWWAN_ERR_QMI_DEVICE_NOT_READY, 822
eQCWWAN_ERR_QMI_DEVICE_STORAGE_FULL, 822
eQCWWAN_ERR_QMI_DISABLED, 822
eQCWWAN_ERR_QMI_ENCODING, 822
eQCWWAN_ERR_QMI_ENVELOPE_CMD_FAILURE, 824

eQCWWAN_ERR_QMI_EVENT_REG_FAILED, 824
eQCWWAN_ERR_QMI_EXTENDED_INTERNAL, 823
eQCWWAN_ERR_QMI_FDN_RESTRICT, 823
eQCWWAN_ERR_QMI_FLOW_SUSPENDED, 822
eQCWWAN_ERR_QMI_GENERAL, 822
eQCWWAN_ERR_QMI_HARDWARE_RESTRICTED, 823
eQCWWAN_ERR_QMI_IFACE, 820
eQCWWAN_ERR_QMI_INCOMPATIBLE_STATE, 823
eQCWWAN_ERR_QMI_INCORRECT_FLOW_FILTER, 822
eQCWWAN_ERR_QMI_INCORRECT_PIN, 821
eQCWWAN_ERR_QMI_INFO_UNAVAILABLE, 823
eQCWWAN_ERR_QMI_INJECT_TIMEOUT, 823
eQCWWAN_ERR_QMI_INSUFFICIENT_RESOURCES, 822
eQCWWAN_ERR_QMI_INTERFACE_NOT_FOUND, 822
eQCWWAN_ERR_QMI_INTERNAL, 821
eQCWWAN_ERR_QMI_INVALID_ARG, 822
eQCWWAN_ERR_QMI_INVALID_CLIENT_ID, 821
eQCWWAN_ERR_QMI_INVALID_DATA_FORMAT, 822
eQCWWAN_ERR_QMI_INVALID_ENVELOPE_CMD, 824
eQCWWAN_ERR_QMI_INVALID_HANDLE, 821
eQCWWAN_ERR_QMI_INVALID_ID, 822
eQCWWAN_ERR_QMI_INVALID_INDEX, 822
eQCWWAN_ERR_QMI_INVALID_IP_FAMILY_PREF, 822
eQCWWAN_ERR_QMI_INVALID_MCAST_HANDLE, 822
eQCWWAN_ERR_QMI_INVALID_MESSAGE_ID, 822
eQCWWAN_ERR_QMI_INVALID_OPERATION, 823
eQCWWAN_ERR_QMI_INVALID_PDP_TYPE, 821
eQCWWAN_ERR_QMI_INVALID_PINID, 821
eQCWWAN_ERR_QMI_INVALID_PROFILE, 821
eQCWWAN_ERR_QMI_INVALID_PROFILE_TYPE, 821
eQCWWAN_ERR_QMI_INVALID_PS_ATTACH_ACTION, 822
eQCWWAN_ERR_QMI_INVALID_QMI_CMD, 823
eQCWWAN_ERR_QMI_INVALID_QOS_ID, 822
eQCWWAN_ERR_QMI_INVALID_REGISTER_ACTION, 822
eQCWWAN_ERR_QMI_INVALID_SERVICE_TYPE, 821
eQCWWAN_ERR_QMI_INVALID_TECH_PREF, 821
eQCWWAN_ERR_QMI_INVALID_TERMINAL_RSP, 824
eQCWWAN_ERR_QMI_INVALID_TRANSITION, 822
eQCWWAN_ERR_QMI_INVALID_TX_ID, 821
eQCWWAN_ERR_QMI_MALFORMED_MSG, 821
eQCWWAN_ERR_QMI_MAX, 823
eQCWWAN_ERR_QMI_MAX_MCAST_REQUESTS_IN_USE, 822
eQCWWAN_ERR_QMI_MAX_QOS_REQUESTS_IN_USE, 822
eQCWWAN_ERR_QMI_MESSAGE_DELIVERY_FAILURE, 822
eQCWWAN_ERR_QMI_MESSAGE_NOT_SENT, 822
eQCWWAN_ERR_QMI_MISSING_ARG, 821
eQCWWAN_ERR_QMI_MSG_BLOCKED, 823
eQCWWAN_ERR_QMI_NETWORK_ABORTED, 823
eQCWWAN_ERR_QMI_NETWORK_NOT_READY, 822
eQCWWAN_ERR_QMI_NETWORK_QOS_UNAWARE, 822
eQCWWAN_ERR_QMI_NO_EFFECT, 821
eQCWWAN_ERR_QMI_NO_ENTRY, 822
eQCWWAN_ERR_QMI_NO_FREE_PROFILE, 821
eQCWWAN_ERR_QMI_NO_MEMORY, 821
eQCWWAN_ERR_QMI_NO_NETWORK_FOUND, 821
eQCWWAN_ERR_QMI_NO_RADIO, 823
eQCWWAN_ERR_QMI_NO_SUBSCRIPTION, 823
eQCWWAN_ERR_QMI_NO_THRESHOLDS, 821
eQCWWAN_ERR_QMI_NOT_A_MCAST_IFACE, 822
eQCWWAN_ERR_QMI_NOT_PROVISIONED, 821
eQCWWAN_ERR_QMI_NOT_SUPPORTED, 823
eQCWWAN_ERR_QMI_OFFSET, 821
eQCWWAN_ERR_QMI_OP_DEVICE_UNSUPPORTED, 821
eQCWWAN_ERR_QMI_OP_NETWORK_UNSUPPORTED, 821
eQCWWAN_ERR_QMI_OP_PARTIAL_FAILURE, 823
eQCWWAN_ERR_QMI_OUT_OF_CALL, 821
eQCWWAN_ERR_QMI_PIN_BLOCKED, 822
eQCWWAN_ERR_QMI_PIN_PERM_BLOCKED, 822
eQCWWAN_ERR_QMI_POLICY_MISMATCH, 823
eQCWWAN_ERR_QMI_REQ, 820
eQCWWAN_ERR_QMI_REQ_SCH, 820
eQCWWAN_ERR_QMI_REQ_TO, 820
eQCWWAN_ERR_QMI_REQUESTED_NUM_UNSUPPORTED, 822
eQCWWAN_ERR_QMI_RSP, 820

eQCWWAN_ERR_QMI_RSP_TO, 820
eQCWWAN_ERR_QMI_SEGMENT_ORDER, 823
eQCWWAN_ERR_QMI_SEGMENT_TOO_LONG, 823
eQCWWAN_ERR_QMI_SESSION_INACTIVE, 822
eQCWWAN_ERR_QMI_SESSION_INVALID, 822
eQCWWAN_ERR_QMI_SESSION_OWNERSHIP, 822
eQCWWAN_ERR_QMI_SIM_FILE_NOT_FOUND, 823
eQCWWAN_ERR_QMI_SIM_NOT_INITIALIZED, 822
eQCWWAN_ERR_QMI_SMSC_ADDR, 823
eQCWWAN_ERR_QMI_SUPS_FAILURE_CAUSE, 823
eQCWWAN_ERR_QMI_TPDU_TYPE, 823
eQCWWAN_ERR_QMI_UNABORTABLE_TRANSACTION, 821
eQCWWAN_ERR_QMI_UNKNOWN, 822
eQCWWAN_ERR_QMI_WIDTH, 824
eQCWWAN_ERR_RESET, 821
eQCWWAN_ERR_SWICM_AM_VERS_ERROR, 823
eQCWWAN_ERR_SWICM_CALL_IN_PROGRESS, 823
eQCWWAN_ERR_SWICM_END, 824
eQCWWAN_ERR_SWICM_FAILED_TO_KILL_SDK_PROCESS, 823
eQCWWAN_ERR_SWICM_INVALID_SESSION_ID, 823
eQCWWAN_ERR_SWICM_INVALID_V4_SESSION_ID, 823
eQCWWAN_ERR_SWICM_INVALID_V6_SESSION_ID, 823
eQCWWAN_ERR_SWICM_NOT_IMPLEMENTED, 823
eQCWWAN_ERR_SWICM_QMI_CLNT_NOT_SUPPORTED, 823
eQCWWAN_ERR_SWICM_QMI_SVC_NOT_SUPPORTED, 823
eQCWWAN_ERR_SWICM_SM_NO_AVAILABLE_SESSIONS, 823
eQCWWAN_ERR_SWICM_SOCKET_IN_USE, 823
eQCWWAN_ERR_SWICM_START, 823
eQCWWAN_ERR_SWICM_TIMEOUT, 823
eQCWWAN_ERR_SWICM_V4DWN_V6DWN, 823
eQCWWAN_ERR_SWICM_V4DWN_V6UP, 823
eQCWWAN_ERR_SWICM_V4UP_V6DWN, 823
eQCWWAN_ERR_SWICM_V4UP_V6UP, 823
eQCWWAN_ERR_SWIDCS_APP_DISCONNECTED, 824
eQCWWAN_ERR_SWIDCS_DEVNODE_NOT_FOUND, 824
eQCWWAN_ERR_SWIDCS_END, 824
eQCWWAN_ERR_SWIDCS_FILEIO_ERR, 824
eQCWWAN_ERR_SWIDCS_IOCTL_ERR, 824
eQCWWAN_ERR_SWIDCS_START, 824
eQCWWAN_ERR_SWIIM_CORRUPTED_FW_IMAGE, 824
eQCWWAN_ERR_SWIIM_END, 824
eQCWWAN_ERR_SWIIM_FILE_NOT_FOUND, 824
eQCWWAN_ERR_SWIIM_FIRMWARE_NOT_DOWNLOADED, 824
eQCWWAN_ERR_SWIIM_FW_ENTER_DOWNLOAD_MODE, 824
eQCWWAN_ERR_SWIIM_FW_FLASH_COMPLETE, 824
eQCWWAN_ERR_SWIIM_FW_INVALID_SLOT_INDEX, 824
eQCWWAN_ERR_SWIIM_FW_PREFERENCE_MISMATCH, 824
eQCWWAN_ERR_SWIIM_FW_SAME_AS_CURRENT_ACTIVE_IMAGE, 824
eQCWWAN_ERR_SWIIM_FW_UPDATE_FAIL, 824
eQCWWAN_ERR_SWIIM_FW_UPDATE_SUCCESS, 824
eQCWWAN_ERR_SWIIM_FW_WAIT_FOR_REBOOT, 824
eQCWWAN_ERR_SWIIM_INVALID_CRASH_STATE, 824
eQCWWAN_ERR_SWIIM_INVALID_PATH, 824
eQCWWAN_ERR_SWIIM_OPENING_DIR, 824
eQCWWAN_ERR_SWIIM_OPENING_FILE, 824
eQCWWAN_ERR_SWIIM_START, 824
eQCWWAN_ERR_SWISM_END, 824
eQCWWAN_ERR_SWISMS_BEARER_DATA_NOT_FOUND, 824
eQCWWAN_ERR_SWISMS_MSG_CORRUPTED, 824
eQCWWAN_ERR_SWISMS_MSG_LEN_TOO_LONG, 824
eQCWWAN_ERR_SWISMS_SMSC_NUM_CORRUPTED, 824
eQCWWAN_ERR_SWISMS_START, 824
eWDS_ERR_PROFILE_REG_3GPP2_ERR_INVALID_IDENT_FOR_PROFILE, 825
eWDS_ERR_PROFILE_REG_3GPP_ACCESS_ERR, 825
eWDS_ERR_PROFILE_REG_3GPP_CONTEXT_NOT_DEFINED, 825
eWDS_ERR_PROFILE_REG_3GPP_ERR_OUT_OF_PROFILES, 825
eWDS_ERR_PROFILE_REG_3GPP_INVAL_PROFILE_FAMILY, 825
eWDS_ERR_PROFILE_REG_3GPP_READ_ONLY_FLAG_SET, 825
eWDS_ERR_PROFILE_REG_3GPP_VALID_FLAG_NOT_SET, 825
eWDS_ERR_PROFILE_REG_END, 825
eWDS_ERR_PROFILE_REG_INVAL_PROFILE_FAMILY, 825

eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_HNDL, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_IDENT, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_OP, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_NUM, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_PROFILE_TYPE, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_INVAL_SUBS_ID, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_LEN_INVALID, 825
eWDS_ERR_PROFILE_REG_RESULT_ERR_LIB_NOT_INITED, 825
eWDS_ERR_PROFILE_REG_RESULT_FAIL, 825
eWDS_ERR_PROFILE_REG_RESULT_LIST_END, 825
qmerrno.h, 818
eQCWWANError, 820
qm_wds_ds_profile_extended_err_codes, 824
qmiSmsMessageList, 349
messageIndex, 350
messageTag, 350
qmiWDSDataBearerTechnology, 350
currentNetwork, 350
ratMask, 350
soMask, 350
qos.h, 825
pack_qos_SLQSQosGetNetworkStatus, 827
pack_qos_SLQSQosSwiReadApnExtraParams, 827
pack_qos_SLQSQosSwiReadDataStats, 828
pack_qos_SLQSSetQosEventCallback, 829
unpack_qos_SLQSQosGetNetworkStatus, 829
unpack_qos_SLQSQosSwiReadApnExtraParams, 830
unpack_qos_SLQSQosSwiReadDataStats, 830
unpack_qos_SLQSSetQosEventCallback, 831
unpack_qos_SLQSSetQosEventCallback_ind, 831
unpack_qos_SLQSSetQosNWStatusCallback_ind, 831
unpack_qos_SLQSSetQosPriEventCallback_ind, 832
unpack_qos_SLQSSetQosStatusCallback_ind, 833
qosDeliveryOrder
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 649
qosFlow
unpack_qos_SLQSQosSwiReadDataStats_t, 507
QosFlowInfo
unpack_qos_SLQSSetQosEventCallback_ind_t, 507

RAN
unpack_nas_GetServingNetwork_t, 460
RAT
nas_QmiNas3GppNetworkRAT, 179
RATMask
currNetworkInfo, 48
wds_currNetworkInfo, 636
RFBandInfoElements, 350
activeBandClass, 350
activeChannel, 350
radioInterface, 350
unpack_nas_GetRFInfo_t, 459
RFTlv
unpack_nas_SetEventReportInd_t, 462
RMAutoConnect
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
RPTlv
NASQmiCbkNasSystemSelPrefInd, 221
RRTlv
unpack_nas_SetEventReportInd_t, 462
RSRPThresListLen
nas_RSRPThresh, 182
RSRQThresListLen
nas_RSRQThresh, 183
RSSIThresListLen
nas_RSSIThresh, 184
RX_EC_IO
nas_NetworkStat1x, 162
RX_PWR
nas_NetworkStat1x, 162
nas_NetworkStatEVDO, 164
RXChan
nas_LTEInfo, 143
rXDroppedCount
unpack_wds_GetPacketStatus_t, 582
rXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 582
rXOkBytesCount
unpack_wds_GetPacketStatus_t, 582
rXPacketErrors
unpack_wds_GetPacketStatus_t, 582
rXPacketOverflows
unpack_wds_GetPacketStatus_t, 582
rXPacketSuccesses
unpack_wds_GetPacketStatus_t, 582
radio
unpack_nas_GetSignalStrengths_t, 461
radio_if
pack_nas_SLQSNasGetTxRxInfo_t, 263
radioIf
nas_ecioListElement, 122
nas_errorRateListElement, 124
nas_rsrqInformation, 182
nas_rxSignalStrengthListElement, 187
RadioIfaces
unpack_dms_GetDeviceCap_t, 403
unpack_dms_GetDeviceCapabilities_t, 404
unpack_nas_GetServingNetwork_t, 460

RadioIfacesSize
unpack_dms_GetDeviceCap_t, 403
unpack_nas_GetServingNetwork_t, 460
radioIfacesSize
unpack_dms_GetDeviceCapabilities_t, 404
radioInterface
nas_RFInfoTlv, 180
nas_roamIndList, 181
nas_servSystem, 190
nas_SignalStrengthTlv, 191
nas_timeInfo, 200
RFBandInfoElements, 350
radioInterfaceList
NASServingSystemInfo, 223
radioInterfaceNo
NASServingSystemInfo, 223
radioInterfaceSize
nas_RFInfoTlv, 180
range
unpack_qos_Port_t, 502
rat
nas_CSGID, 115
nas_MNRInfo, 159
ratMask
qmiWDSDataBearerTechnology, 350
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
rawHorConfidence
pack_loc_SLQSLOCInjectPosition_t, 246
rawHorUncCircular
pack_loc_SLQSLOCInjectPosition_t, 246
rawLen
uim_fileAttributes, 388
rawValue
uim_fileAttributes, 388
readTransparent
pack_uim_ReadTransparent_t, 299
Reason
pack_voice_SLQSVoiceGetCallForwardingStatus-_t, 316
pack_voice_SLQSVoiceSetCallBarringPassword_t, 323
reason
pack_voice_SLQSVoiceGetCallBarring_t, 316
pack_voice_SLQSVoiceSetSUPSService_t, 327
unpack_qos_SLQSSetQosStatusCallback_ind_t, 509
voice_ccSUPSType, 616
receiptAction
sms_routeEntry, 364
reconfigReqd
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 592
recordCount
uim_fileAttributes, 388
recordSize
uim_fileAttributes, 388
RedirPartyNum
voice_arrRedirPartyNum, 606
refData
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 425
refString
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 425
refpn
nas_CDMAInfo, 106
refreshComplete
pack_uim_SLQSUIMRefreshComplete_t, 305
regAction
pack_nas_SLQSInitiateNetworkRegistration_t, 258
RegForeignNID
unpack_nas_GetCDMANetworkParameters_t, 455
RegForeignSID
unpack_nas_GetCDMANetworkParameters_t, 455
RegHomeSID
unpack_nas_GetCDMANetworkParameters_t, 455
regPrd
nas_AddCDMASysInfo, 102
regRefresh
pack_uim_SLQSUIMRefreshRegister_t, 306
regRejectInfoValid
nas_GSMSysInfo, 132
nas_LTESysInfo, 156
nas_WCDMASysInfo, 212
regState
nas_servSystem, 190
registerFlag
uim_registerRefresh, 394
RegistrationState
unpack_nas_GetServingNetwork_t, 460
registrationState
NASServingSystemInfo, 223
rejCause
nas_GSMSysInfo, 132
nas_LTESysInfo, 156
nas_WCDMASysInfo, 213
rejectCause
nas_RejectReasonTlv, 180
rejectSrvDomain
nas_GSMSysInfo, 132
nas_LTESysInfo, 156
nas_WCDMASysInfo, 213
reliabilityClass
LibPackGPRSRequestedQoS, 59
wds_GPRSQoS, 642
remPartyNumber
voice_remotePartyNum, 632
RemotePartyName
voice_getAllCallRmtPtyName, 625
RemotePartyNum
voice_getAllCallRmtPtyNum, 626
resBerRatio
LibPackUMTSQoS, 86
wds_UMTSMinQoS, 649
ResCode
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 425
resetInfoInd

pack_dms_SLQSDmsSwiIndicationRegister_t, 228
resultcode
_unpack_swiavms_SLQSAvMSEventReportInd_t, 31
_unpack_swiavms_SLQSAvMSSessionGetInfo_t, 32
unpack_swiavms_SLQSAvMSGetSettings_t, 530
unpack_swiavms_SLQSAvMSSendSelection_t, 530
unpack_swiavms_SLQSAvmsSetEventReport_t, 531
unpack_swiavms_SLQSAvMSSetSettings_t, 531
unpack_swiavms_SLQSAvMSStartSession_t, 531
unpack_swiavms_SLQSAvMSStopSession_t, 532
RetryCount
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
revTunneling
unpack_wds_GetMobileIPProfile_t, 580
ReverseMac
nas_protocolSubtypeElement, 177
RfSarState
pack_sar_SLQSSetRfSarState_t, 280
RmTrasnferStaticsReq
pack_wds_RMSetTransferStatistics_t, 331
rmTrasnferStaticsReq, 351
bResetStatistics, 351
ulMask, 351
rms.h, 833
pack_rms_GetSMSWake, 834
pack_rms_SetSMSWake, 834
unpack_rms_GetSMSWake, 834
unpack_rms_SetSMSWake, 835
RmtPtyNum
voice_arrRemotePartyNum, 607
roamIndicator
nas_roamIndList, 181
RoamIndicatorVal
unpack_nas_SLQSGetServingSystem_t, 471
roamOrigVoiceSO
voice_prefVoiceSO, 630
RoamPref
NASRoamPreferenceTlv, 221
roamStatus
nas_sysInfoCommon, 196
roamStatusValid
nas_sysInfoCommon, 196
roamTimerValue
voice_roamTimer, 633
Roaming
nas_QmiNas3GppNetworkInfo, 178
unpack_nas_GetCDMANetworkParameters_t, 455
unpack_nas_GetServingNetwork_t, 460
roaming
unpack_nas_SetRoamingIndicatorCallback_ind_t, 465
RoamingIndicatorList
unpack_nas_SLQSGetServingSystem_t, 471
routeList
sms_setRoutesReq, 368
routeStorage
sms_routeEntry, 364
rptRate
nas_LTESigRptCfg, 151
nas_LTESigRptConfig, 152
rscp
nas_rxInfo, 185
nas_UMTSInfo, 203
tdscdmaSigInfoExt, 374
rsrp
lteSSInfo, 100
nas_cellParams, 112
nas_rxInfo, 185
nas_RxSigInfo, 186
nas_umtsLTENbrCell, 205
rsrplevel
nas_lteRsrpinformation, 149
rsrq
lteSSInfo, 100
nas_cellParams, 112
nas_rsrqInformation, 183
nas_SccRxInfo, 187
nas_umtsLTENbrCell, 205
rsrqDelta
nas_SLQSSignalStrengthsIndReq, 192
rsrqInfo
nas_SLQSSignalStrengthsInformation, 193
unpack_nas_SLQSGetSignalStrength_t, 472
rssi
cdmaSSInfo, 46
hdrSSInfo, 58
lteSSInfo, 100
nas_cellParams, 112
nas_gsmCellInfo, 128
tdscdmaSigInfoExt, 374
unpack_nas_GetSignalStrengths_t, 461
rx_bytes
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
rx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
rxChainIndex
nas_RxSigInfo, 186
RxDropConutTlv
unpack_RMTransferStatistics_ind_t, 520
rxLev
nas_GERANInfo, 126
rxOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 588
RxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 520
RxOkConutTlv
unpack_RMTransferStatistics_ind_t, 520
rxPower
nas_rxInfo, 185

nas_RxSigInfo, 186
RxQFilter
unpack_qos_QosFlowInfo_t, 503
RxQFlowGranted
unpack_qos_QosFlowInfo_t, 503
rxSignalStrength
nas_rxSignalStrengthListElement, 187
rxSignalStrengthDelta
nas_SLQSSignalStrengthsIndReq, 192
rxSignalStrengthInfo
nas_SLQSSignalStrengthsInformation, 193
rxSignalStrengthList
unpack_nas_SLQSGetSignalStrength_t, 472
rxSignalStrengthListLen
unpack_nas_SLQSGetSignalStrength_t, 472

SCI
unpack_nas_GetCDMANetworkParameters_t, 455
SCM
unpack_nas_GetCDMANetworkParameters_t, 455
SDPTlv
NASQmiCbkNasSystemSelPrefInd, 221
SDU_HDR_LEN
common.h, 655
SHORT
SwiDataTypes.h, 878
SI
voice_callFWExtInfo, 613
sIntraSearch
nas_LTEInfoIntrafreq, 146
SLQSSSTlv
unpack_nas_SetEventReportInd_t, 462
SMS_CONFIG_LEN
sms.h, 840
SMS_MAX_SMS_ROUTES
sms.h, 840
SMS_NUM_OF_SET
sms.h, 840
SMSC_TYPE_LEN
sms.h, 840
sMSCAddress, 369
data, 369
length, 369
sMSCAddressInfo
sms.h, 841
sMSCAddressTlv, 369
SMSCInfo, 369
TlvPresent, 369
SMSCInfo
sMSCAddressTlv, 369
SMSCTlv
unpack_sms_SetNewSMSCallback_ind_t, 523
sMSEtwsMessage, 369
data, 370
length, 370
notificationType, 370
sMSEtwsMessageInfo
sms.h, 841
sMSEtwsMessageTlv, 370
EtwsMessageInfo, 370
TlvPresent, 370
sMSEtwsPlmn, 371
mobileCountryCode, 371
mobileNetworkCode, 371
sMSEtwsPlmnInfo
sms.h, 841
sMSMTMessage, 371
messageIndex, 372
storageType, 372
sMSMTMessageInfo
sms.h, 841
sMSMessageMode, 371
messageMode, 371
sMSMessageModeInfo
sms.h, 841
sMSOnIMS, 372
smsOnIMS, 372
sMSOnIMSInfo
sms.h, 841
sMSOnIMSTlv, 372
IMSInfo, 372
TlvPresent, 372
SMSSupport
pack_dms_SetCustFeature_t, 226
unpack_dms_GetCustFeature_t, 402
sMSTransferRouteMTMessage, 373
ackIndicator, 373
data, 373
format, 373
length, 373
transactionID, 373
sMSTransferRouteMTMessageInfo
sms.h, 842
SNR
nas_NetworkStatEVDO, 164
sNonIntraSearch
nas_LTEInfoIntrafreq, 146
SO
nas_NetworkStat1x, 162
SOMask
currNetworkInfo, 48
wds_currNetworkInfo, 636
SPC_SIZE
dms.h, 665
sPhyCaAggPcellInfo
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 463
sPhyCaAggScellDlBw
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 463
sPhyCaAggScellIndType
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 463
sPhyCaAggScellIndex
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 463
sPhyCaAggScellInfo

unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 463
SSInfo
unpack_nas_SetServingSystemCallback_ind_t, 465
sSLQSSignalStrengthsInfo
nas_SLQSSignalStrengthsTlv, 193
SSTlv
unpack_nas_SetEventReportInd_t, 462
SUPSType
pack_voice_SLQSVoiceManageCalls_t, 321
SWI_API
SwiDataTypes.h, 877
SWIWWANCMAPI.h, 889
sar.h, 835
pack_sar_SLQSGetRfSarState, 835
pack_sar_SLQSSetRfSarState, 836
unpack_sar_SLQSGetRfSarState, 836
unpack_sar_SLQSSetRfSarState, 836
scell_idx
nas_PhyCaAggScellIndex, 169
NASPhyCaAggScellIndex, 218
scell_state
nas_PhyCaAggScellIndType, 169
nas_PhyCaAggScellInfo, 172
NASPhyCaAggScellIndType, 218
NASPhyCaAggScellInfo, 219
screeningInd
voice_connectNumInfo, 621
sduErrorRatio
LibPackUMTSQoS, 87
wds_UMTSMinQoS, 649
seDNSIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
seDNSIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
sePCSCFIPv4Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
sePCSCFIPv6Address
unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 597
secActivate
uim_fileAttributes, 388
secActivateMask
uim_fileAttributes, 388
secChA
nas_CDMAChannel, 105
secChB
nas_CDMAChannel, 105
secDeactivate
uim_fileAttributes, 388
secDeactivateMask
uim_fileAttributes, 388
secIncrease
uim_fileAttributes, 388
secIncreaseMask
uim_fileAttributes, 388
SecProt
nas_protocolSubtypeElement, 177
secRead
uim_fileAttributes, 388
secReadMask
uim_fileAttributes, 388
secWrite
uim_fileAttributes, 388
secWriteMask
uim_fileAttributes, 388
secdns
unpack_wds_GetDefaultProfile_t, 577
secdnsv6
unpack_wds_GetDefaultProfile_t, 577
second
nas_timeInfo, 200
nas_UniversalTime, 206
secondaryDNS
pack_wds_SetDefaultProfile_t, 332
SecondaryDNSV4
unpack_wds_SLQSGetRuntimeSettings_t, 590
SecondaryDNSV6
unpack_wds_SLQSGetRuntimeSettings_t, 591
secondaryHA
unpack_wds_GetMobileIPProfile_t, 580
SectorIDLen
nas_NetworkStatEVDO, 164
selNetwork
nas_servSystem, 190
selected
sms_BroadcastConfig, 356
sms_CDMABroadcastConfig, 356
selectedNetwork
NASServingSystemInfo, 223
selection
pack_swiavms_SLQSAvMSSendSelection_t, 290
pack_swioma_SLQSOMADMSendSelection_t, 296
sensorData_t, 351
flags, 352
sensorDataLen, 352
timeOfFirstSample, 352
timeOffset, 352
xAxis, 352
yAxis, 352
zAxis, 352
sensorDataLen
sensorData_t, 352
ServerAddrList
unpack_wds_SLQSGetRuntimeSettings_t, 591
serviceCategory
sms_CDMABroadcastConfig, 356
serviceDomain
nas_RejectReasonTlv, 180
servingCellId
nas_LTEInfoIntrafreq, 146

ServingSystem
unpack_nas_SLQSGetServingSystem_t, 471
sessionEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 592
SessionId
pack_loc_Start_t, 251
pack_loc_Stop_t, 251
sessionId
unpack_loc_PositionRpt_Ind_t, 450
sessionInfo
pack_uim_ChangePin_t, 298
pack_uim_ReadTransparent_t, 299
pack_uim_SetPinProtection_t, 300
pack_uim_SLQSUIMAuthenticate_t, 301
pack_uim_SLQSUIMGetFileAttributes_t, 303
pack_uim_SLQSUIMRefreshComplete_t, 305
pack_uim_SLQSUIMRefreshGetLastEvent_t, 305
pack_uim_SLQSUIMRefreshOK_t, 306
pack_uim_SLQSUIMRefreshRegister_t, 306
pack_uim_UnblockPin_t, 308
pack_uim_VerifyPin_t, 309
SessionInfoConfig
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 534
SessionInfoFota
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 534
SessionInfoNotification
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 534
sessionResponse
unpack_swiavms_SLQSAvMSStartSession_t, 532
SessionState
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
sessionStatus
unpack_loc_PositionRpt_Ind_t, 450
unpack_omaDmNotificationsTlv_t, 497
SessionType
pack_swioma_SLQSOMADMGetSessionInfo_t, 295
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
sessionType
pack_swiavms_SLQSAvMSStartSession_t, 292
pack_swiavms_SLQSAvMSStopSession_t, 293
pack_swioma_SLQSOMADMCancelSession_t, 295
pack_swioma_SLQSOMADMStartSession_t, 297
uim_refreshevent, 393
uim_sessionInformation, 395
uim_UIMSessionInformation, 399
unpack_omaDmFotaTlv_t, 496
set_fix_rate
pack_swiloc_SwiLocSetAutoStart_t, 294
set_fix_type
pack_swiloc_SwiLocSetAutoStart_t, 294
set_function
pack_swiloc_SwiLocSetAutoStart_t, 294
set_max_dist
pack_swiloc_SwiLocSetAutoStart_t, 294
set_max_time
pack_swiloc_SwiLocSetAutoStart_t, 294
Severity
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
severity
unpack_omaDmFotaTlv_t, 496
shortName
nas_PLMNNetworkNameData, 175
unpack_nas_SLQSGetPLMNName_t, 468
shortNameCI
unpack_nas_SLQSGetPLMNName_t, 468
shortNameEn
unpack_nas_SLQSGetPLMNName_t, 469
shortNameLen
nas_PLMNNetworkNameData, 175
unpack_nas_SLQSGetPLMNName_t, 469
shortNameSB
unpack_nas_SLQSGetPLMNName_t, 469
shortNameSpareBits
nas_PLMNNetworkNameData, 175
sid
nas_CDMAInfo, 106
nas_sidNid, 190
unpack_nas_GetHomeNetwork_t, 457
SidNid
nas_homeSIDNID, 137
SigInd
LibPackUMTSReqQoSSigInd, 87
sigInfo
nas_SccRxInfo, 188
signalStrength
nas_SignalStrengthTlv, 191
signalStrengthReqMask
unpack_nas_SLQSGetSignalStrength_t, 472
SimCapability
unpack_dms_GetDeviceCap_t, 403
simCapability
unpack_dms_GetDeviceCapabilities_t, 404
sinr
hdrSSInfo, 58
nas_SLQSSignalStrengthsInformation, 193
tdscdmaSigInfoExt, 374
unpack_nas_SLQSGetSignalStrength_t, 472
sinrDelta
nas_SLQSSignalStrengthsIndReq, 192
sinrThresholdList
nas_SLQSSignalStrengthsIndReq, 192
sinrThresholdListLen
nas_SLQSSignalStrengthsIndReq, 192
sku_str
unpack_dms_GetFirmwareInfo_t, 406
slot
pack_uim_SLQSUIMPowerDown_t, 304

pack_uim_SLQSUIMPowerUp_t, 304
slot_t, 352
bICCID, 353
bICCIDLength, 353
bLogicalSlot, 353
uPhyCardStatus, 353
uPhySlotStatus, 353
slotInf, 353
AppStatus, 355
cardState, 355
errorState, 355
numApp, 355
upinRetries, 355
upinState, 355
upukRetries, 355
SlotInfo
uim_cardStatus, 382
slots_t, 355
uimSlotStatus, 355
slotsstatusChange
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 542
SlqsProfile3GPP
unpackWdsProfileParam, 598
wds_profileInfo, 646
SlqsProfile3GPP2
unpackWdsProfileParam, 598
wds_profileInfo, 646
sms.h
LITEQMI_QMI_CBK_PARAM_NOCHANGE, 842
LITEQMI_QMI_CBK_PARAM_RESET, 842
LITEQMI_QMI_CBK_PARAM_SET, 842
sms.h, 837
eqmiCbkSetStatus, 842
MAX_MSE_TWS_MSG, 840
MAX_SMS_LIST_SIZE, 840
pack_sms_GetSMSCAddress, 842
pack_sms_SLQSDeleteSMS, 844
pack_sms_SLQSGetIndicationRegister, 844
pack_sms_SLQSGetMessageWaiting, 845
pack_sms_SLQSGetSMS, 845
pack_sms_SLQSGetSMSList, 846
pack_sms_SLQSGetSmsBroadcastConfig, 845
pack_sms_SLQSGetTransLayerInfo, 846
pack_sms_SLQSGetTransNWRegInfo, 847
pack_sms_SLQSModifySMSStatus, 847
pack_sms_SLQSSendAsyncSMS, 847
pack_sms_SLQSSetIndicationRegister, 848
pack_sms_SLQSSetSmsBroadcastActivation, 848
pack_sms_SLQSSetSmsBroadcastConfig, 848
pack_sms_SLQSSetSmsStorage, 849
pack_sms_SLQSSmsGetMaxStorageSize, 849
pack_sms_SLQSSmsGetMessageProtocol, 849
pack_sms_SLQSSmsSetRoutes, 850
pack_sms_SLQSSwiGetSMSStorage, 850
pack_sms_SaveSMS, 842
pack_sms_SendSMS, 843
pack_sms_SetNewSMSCallback, 843
pack_sms_SetSMSCAddress, 844
SMS_CONFIG_LEN, 840
SMS_MAX_SMS_ROUTES, 840
SMS_NUM_OF_SET, 840
SMSC_TYPE_LEN, 840
sMSCAddressInfo, 841
sMSEtwsMessageInfo, 841
sMSEtwsPlmnInfo, 841
sMSMTMessageInfo, 841
sMSMessageModeInfo, 841
sMSOnIMSInfo, 841
sMSTransferRouteMTMessageInfo, 842
unpack_sms_GetSMSCAddress, 851
unpack_sms_SLQSDeleteSMS, 853
unpack_sms_SLQSGetIndicationRegister, 853
unpack_sms_SLQSGetMessageWaiting, 853
unpack_sms_SLQSGetSMS, 854
unpack_sms_SLQSGetSMSList, 854
unpack_sms_SLQSGetSmsBroadcastConfig, 854
unpack_sms_SLQSGetTransLayerInfo, 855
unpack_sms_SLQSGetTransNWRegInfo, 855
unpack_sms_SLQSModifySMSStatus, 855
unpack_sms_SLQSSendAsyncSMS, 856
unpack_sms_SLQSSetIndicationRegister, 856
unpack_sms_SLQSSetSmsBroadcastActivation, 856
unpack_sms_SLQSSetSmsBroadcastConfig, 857
unpack_sms_SLQSSetSmsStorage, 857
unpack_sms_SLQSSmsGetMaxStorageSize, 857
unpack_sms_SLQSSmsGetMessageProtocol, 858
unpack_sms_SLQSSmsSetRoutes, 858
unpack_sms_SLQSSwiGetSMSStorage, 858
unpack_sms_SLQSWmsMemoryFullCallBack_ind, 859
unpack_sms_SaveSMS, 851
unpack_sms_SendSMS, 851
unpack_sms_SetNewSMSCallback, 852
unpack_sms_SetNewSMSCallback_ind, 852
unpack_sms_SetSMSCAddress, 852
sms_BroadcastConfig, 355
fromServiceId, 356
selected, 356
toServiceId, 356
sms_CDMABroadcastConfig, 356
language, 356
selected, 356
serviceCategory, 356
sms_getIndicationReg, 356
pRegCallStatInfoEvt, 357
pRegTransLayerInfoEvt, 357
pRegTransNWRegInfoEvt, 357
sms_getMsgWaitingInfo, 357
msgWaitInfo, 358
numInstances, 358
sms_getTransLayerInfo, 358
pRegInd, 358
pTransLayerInfo, 358
sms_getTransNWRegInfo, 359

pRegStatus, 359
sms_maxStorageSizeReq, 359
pMessageMode, 360
storageType, 360
sms_maxStorageSizeResp, 360
freeSlots, 360
maxStorageSize, 360
sms_messageWaitingInfoContent, 360
activeInd, 361
msgCount, 361
msgType, 361
sms_msgProtocolResp, 361
msgProtocol, 361
sms_qaQmi3GPP2BroadcastCfgInfo, 361
activated_ind, 362
CDMABroadcastConfig, 362
num_instances, 362
sms_qaQmi3GPPBroadcastCfgInfo, 362
activated_ind, 363
broadcastConfig, 363
num_instances, 363
sms_routeEntry, 363
messageClass, 364
messageType, 364
receiptAction, 364
routeStorage, 364
sms_sendAsyncsmsParams, 364
messageFormat, 366
messageSize, 366
pFollowOnDC, 366
pForceOnDC, 366
pLinktimer, 366
pMessage, 366
pRetryMessage, 366
pRetryMessageId, 366
pServiceOption, 366
pSmsOnIms, 366
pUserData, 366
sms_setIndicationReg, 366
pRegCallStatInfoEvt, 367
pRegTransLayerInfoEvt, 367
pRegTransNWRegInfoEvt, 367
sms_setRoutesReq, 367
numOfRoutes, 368
pTransferStatusReport, 368
routeList, 368
sms_transLayerInfo, 368
TransCap, 368
TransType, 368
smsOnIMS
sMSOnIMS, 372
smsStorage
pack_sms_SLQSSetSmsStorage_t, 288
snr
loc_satelliteInfo, 96
lteSSInfo, 100
nas_SccRxInfo, 188
snrlevel
nas_lteSnrinformation, 152
soMask
qmiWDSDataBearerTechnology, 350
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
Source
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
source
altSrcInfo_t, 41
unpack_dms_GetNetworkTime_t, 409
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 416
unpack_dms_SLQSDmsSwiGetResetInfo_t, 417
sourceIPMask
LibPackTFTIDParams, 84
SourceLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
spc
pack_dms_ResetToFactoryDefaults_t, 225
pack_nas_SetACCOLC_t, 252
pack_wds_SetMobileIPProfile_t, 335
spn
nas_serviceProviderName, 188
unpack_nas_SLQSGetPLMNName_t, 469
spnEncoding
unpack_nas_SLQSGetPLMNName_t, 469
spnLength
nas_serviceProviderName, 188
unpack_nas_SLQSGetPLMNName_t, 469
srcPortRangeEnd
LibPackTFTIDParams, 84
srcPortRangeStart
LibPackTFTIDParams, 84
srvCapability
nas_detailSvcInfo, 120
nas_sysInfoCommon, 196
srvCapabilityValid
nas_sysInfoCommon, 196
srvDomain
nas_sysInfoCommon, 196
SrvDomainPref
NASServDomainPrefTlv, 221
srvDomainValid
nas_sysInfoCommon, 196
srvOption
voice_arrSvcOption, 608
srvStatus
nas_detailSvcInfo, 120
nas_GSMSrvStatusInfo, 129
nas_SrvStatusInfo, 194
srxlev
nas_cellParams, 112
nas_gsmCellInfo, 128
nas_umtsLTENbrCell, 205
nas_wcdmaCellInfo, 207
stage

uim_refreshevent, 393
State
nas_NetworkStat1x, 162
nas_NetworkStatEVDO, 164
state
pack_loc_SLQSLOCSetCradleMountConfig_t, 249
unpack_dms_GetActivationState_t, 401
unpack_dms_SLQSUIMGetState_t, 429
unpack_omaDmConfigTlv_t, 494
unpack_omaDmFotaTlv_t, 496
unpack_qos_QosFlowInfoState_t, 504
statmask
pack_wds_GetPacketStatus_t, 330
StatsMask
transferStatInd, 376
statsMask
wds_TrStatInd, 646
StatsPeriod
transferStatInd, 376
statsPeriod
wds_TrStatInd, 646
Status
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
status
nas_lteEARFCN, 140
nas_ltePCI, 148
nas_wcdmaUARFCN, 213
pack_sms_SetNewSMSCallback_t, 282
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_DeleteAssistData_Ind_t, 442
unpack_loc_SetExtPowerConfig_Ind_t, 451
unpack_loc_SetOperationMode_Ind_t, 452
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 508
unpack_qos_SLQSSetQosStatusCallback_ind_t, 509
storageIndex
FMSImageIdElement, 55
storageType
pack_sms_SaveSMS_t, 281
pack_sms_SLQSDeleteSMS_t, 283
pack_sms_SLQSGetSMS_t, 284
pack_sms_SLQSGetSMSList_t, 285
pack_sms_SLQSModifySMSStatus_t, 286
sms_maxStorageSizeReq, 360
sMSMTMessage, 372
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 529
Streaming Download Protocol (lite-fw), 24
String
unpack_dms_GetDeviceHardwareRev_t, 404
unpack_dms_GetDeviceMfr_t, 404
unpack_dms_GetFSN_t, 407
unpack_dms_UIMGetICCID_t, 430
stringSize
unpack_dms_GetDeviceHardwareRev_t, 404
unpack_dms_GetDeviceMfr_t, 404
unpack_dms_UIMGetICCID_t, 431
subAddr
voice_calledPartySubAdd, 610
subAddrLen
voice_calledPartySubAdd, 610
subAddrType
voice_calledPartySubAdd, 610
subnetMask
unpack_qos_IPv4Addr_t, 500
SubnetMaskV4
unpack_wds_SLQSGetRuntimeSettings_t, 591
subsType
pack_voice_SLQSVoiceBindSubscription_t, 312
SupUSBComps
unpack_dms_GetUSBComp_t, 412
SuppOA
voice_CUGInfo, 622
SuppPrefCUG
voice_CUGInfo, 622
svInfoMask
loc_satelliteInfo, 96
svListLen
loc_satelliteInfo, 96
svStatus
loc_satelliteInfo, 96
svc
pack_qmi_t, 277
SvcClass
voice_callFWExtInfo, 613
voice_callFWInfo, 614
SvcStatus
voice_callFWExtInfo, 613
voice_callFWInfo, 614
svcType
voice_ccSUPSType, 616
sw1
uim_cardResult, 381
sw2
uim_cardResult, 381
swVerString
pack_dms_SLQSSwiSetHostDevInfo_t, 230
unpack_dms_SLQSSwiGetHostDevInfo_t, 426
SwiDataTypes.h, 877
BOOL, 878
BYTE, 878
CHAR, 878
FLOAT, 878
INT32, 878
INT8, 878
LPCSTR, 878
SHORT, 878
SWI_API, 877
ULONG, 878
ULONGLONG, 878
UNUSEDPARAM, 877
USHORT, 878
WORD, 878
swiavms.h, 859

pack_swiavms_SLQSAvMSGetSettings, 868
pack_swiavms_SLQSAvMSSendSelection, 869
pack_swiavms_SLQSAvMSSessionGetInfo, 869
pack_swiavms_SLQSAvMSSetSettings, 870
pack_swiavms_SLQSAvMSSetSettingsNoAutoRebootField, 871
pack_swiavms_SLQSAvMSStartSession, 872
pack_swiavms_SLQSAvMSStopSession, 872
pack_swiavms_SLQSAvmsSetEventReport, 870
PackSwiAVMSSettingsAPNInfo, 862
PackSwiAVMSSettingsConnectionRetryTimers, 862
PackSwiAVMSSettingsPeriodsInfo, 862
PackSwiAvmsSetSettingsAPNInfo, 862
PackSwiAvmsSetSettingsConnectionRetryTimers, 862
PackSwiAvmsSetSettingsPeriodInfo, 862
unpack_swiavms_SLQSAvMSEventReportInd, 873
unpack_swiavms_SLQSAvMSEventReportInd_t, 862
unpack_swiavms_SLQSAvMSGetSettings, 873
unpack_swiavms_SLQSAvMSSendSelection, 874
unpack_swiavms_SLQSAvMSSessionGetInfo, 874
unpack_swiavms_SLQSAvMSSessionGetInfo_t, 863
unpack_swiavms_SLQSAvMSSetSettings, 875
unpack_swiavms_SLQSAvMSStartSession, 876
unpack_swiavms_SLQSAvMSStopSession, 876
unpack_swiavms_SLQSAvmsSetEventReport, 875
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 863
UnpackSwiAvmsEventReportConfig, 864
UnpackSwiAvmsEventReportConnectionRequest, 864
UnpackSwiAvmsEventReportDataSessionStatus, 865
UnpackSwiAvmsEventReportHTTPStatus, 865
UnpackSwiAvmsEventReportNotification, 866
UnpackSwiAvmsEventReportPackageID, 867
UnpackSwiAvmsEventReportRegStatus, 867
UnpackSwiAvmsEventReportSessionType, 867
UnpackSwiAvmsEventReportWAMSParamChange, 868
swiloc.h, 878
pack_swiloc_SwiLocGetAutoStart, 878
pack_swiloc_SwiLocSetAutoStart, 879
unpack_swiloc_SwiLocGetAutoStart, 879
unpack_swiloc_SwiLocSetAutoStart, 879
swioma.h, 880
pack_swioma_SLQSOMADMAlertCallback, 881
pack_swioma_SLQSOMADMCancelSession, 881
pack_swioma_SLQSOMADMGetSessionInfo, 882
pack_swioma_SLQSOMADMGetSettings, 883
pack_swioma_SLQSOMADMSendSelection, 883
pack_swioma_SLQSOMADMSetSettings, 884
pack_swioma_SLQSOMADMStartSession, 884
unpack_swioma_SLQSOMADMAlertCallback, 885
unpack_swioma_SLQSOMADMAlertCallback_ind, 886
unpack_swioma_SLQSOMADMCancelSession, 886
unpack_swioma_SLQSOMADMGetSessionInfo, 886
unpack_swioma_SLQSOMADMGetSettings, 887
unpack_swioma_SLQSOMADMSendSelection, 887
unpack_swioma_SLQSOMADMSetSettings, 888
unpack_swioma_SLQSOMADMStartSession, 888
switchOption
pack_voice_SLQSVoiceALSSetLineSwitching_t, 311
sysInfoCDMA
nas_CDMASysInfo, 111
sysInfoGSM
nas_GSMSysInfo, 132
sysInfoHDR
nas_HDRSysInfo, 137
sysInfoLTE
nas_LTESysInfo, 156
sysInfoWCDMA
nas_WCDMASysInfo, 213
system
loc_satelliteInfo, 96
loc_SV, 98
SystemID
unpack_nas_SLQSGetServingSystem_t, 471
systemID
nas_CDMASysInfo, 111
systemMode
nas_CommInfo, 114
szAPN
_PackSwiAvmsSetSettingsAPNInfo, 28
szAlertMsg
_UnpackSwiAvmsEventReportConfig, 35
szCarrier_str
_litefw_FirmwareInfo_, 27
szCarrierPriversion_str
_litefw_FirmwareInfo_, 28
szDescription
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 33
szFwversion_str
_litefw_FirmwareInfo_, 28
szModelid_str
_litefw_FirmwareInfo_, 28
szName
_UnpackSwiAvmsEventReportBinaryUpdateSessionInfo, 33
szPWD
_PackSwiAvmsSetSettingsAPNInfo, 28
szPackageid_str
_litefw_FirmwareInfo_, 28
szSku_str
_litefw_FirmwareInfo_, 28
szUname

_PackSwiAvmsSetSettingsAPNInfo, 28
szVersion
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34

t3396_apn
unpack_nas_SLQSNasTimerCallback_ind_t, 486
t3396_plmn_id
unpack_nas_SLQSNasTimerCallback_ind_t, 486
t3396_val
unpack_nas_SLQSNasTimerCallback_ind_t, 486
TCPDstPort
unpack_qos_swiQosFilter_t, 513
TCPSrcPort
unpack_qos_swiQosFilter_t, 513
TDSCDMAECIOThreshListLen
nas_TDSCDMAECIOThresh, 196
TDSCDMARSCPThreshListLen
nas_TDSCDMARSCPThresh, 197
TDSCDMARSSIThreshListLen
nas_TDSCDMARSSIThresh, 197
TDSCDMASINRThreshListLen
nas_TDSCDMASINRThresh, 198
TRMessageTlv
unpack_sms_SetNewSMSCallback_ind_t, 524
TX_PWR
nas_NetworkStat1x, 162
TXChan
nas_LTEInfo, 143
tXDroppedCount
unpack_wds_GetPacketStatus_t, 582
tXOKBytesLastCall
unpack_wds_GetPacketStatus_t, 582
tXOkBytesCount
unpack_wds_GetPacketStatus_t, 582
tXPacketErrors
unpack_wds_GetPacketStatus_t, 582
tXPacketOverflows
unpack_wds_GetPacketStatus_t, 582
tXPacketSuccesses
unpack_wds_GetPacketStatus_t, 582
Tables, 25
tac
nas_LTEInfoIntrafreq, 146
nas_LTESysInfo, 156
tacValid
nas_LTESysInfo, 156
TdsBandCapability
unpack_dms_SLQSGetBandCapability_t, 421
tdscdmaSigInfoExt, 373
ecio, 374
rscp, 374
rssi, 374
sinr, 374
techName
unpack_wds_SLQSSetPacketSrvStatusCallback_-t, 592
Technology
nas_DeviceConfigDetail, 121
unpack_wds_SLQSGetRuntimeSettings_t, 591
TechnologyPref
pack_nas_SetNetworkPreference_t, 254
tempData_t, 374
temperature, 375
temperatureDataLen, 375
timeOfFirstSample, 375
timeOffset, 375
timeSource, 375
temperature
nas_CommInfo, 114
tempData_t, 375
temperatureDataLen
tempData_t, 375
threshGsmHigh
nas_lteGsmCellInfo, 141
threshGsmLow
nas_lteGsmCellInfo, 141
threshServingLow
nas_LTEInfoIntrafreq, 146
threshXHigh
nas_infoInterFreq, 139
threshXLow
nas_infoInterFreq, 139
threshXhigh
nas_lteWcdmaCellInfo, 157
threshXlow
nas_lteWcdmaCellInfo, 157
Time
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
time
NASTimeInfoTlv, 223
TimeLength
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
timeMsec
pack_loc_SLQSLOCInjectUTCTime_t, 249
timeOfFirstSample
sensorData_t, 352
tempData_t, 375
timeOffset
sensorData_t, 352
tempData_t, 375
timeSource
tempData_t, 375
timeTlv
NASQmiCbkNasSwiOTAMessageInd, 220
timeUncMsec
pack_loc_SLQSLOCInjectUTCTime_t, 249
timeZone
nas_timeInfo, 200
timeout
pack_pds_SetPDSDefaults_t, 276
pack_qmi_t, 277
Timers
_PackSwiAVMSSettingsConnectionRetryTimers, 30

_PackSwiAvmsSetSettingsConnectionRetry-Timers, 29
timestamp
unpack_dms_GetNetworkTime_t, 409
timestampAge
pack_loc_SLQSLOCInjectPosition_t, 246
timestampUtc
pack_loc_SLQSLOCInjectPosition_t, 246
timingAdvance
nas_GERANInfo, 126
TlvPresent
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
_UnpackSwiAvmsEventReportConfig, 35
_UnpackSwiAvmsEventReportConnectionRequest, 35
_UnpackSwiAvmsEventReportDataSessionStatus, 36
_UnpackSwiAvmsEventReportHTTPStatus, 37
_UnpackSwiAvmsEventReportNotification, 38
_UnpackSwiAvmsEventReportPackageID, 39
_UnpackSwiAvmsEventReportRegStatus, 39
_UnpackSwiAvmsEventReportSessionType, 40
_UnpackSwiAvmsEventReportWAMSParam-Change, 40
dms_ActivationStatusTlv, 49
dms_OperatingModeTlv, 50
eTWSPLMNInfoTlv, 54
messageModeTlv, 100
nas_PhyCaAggPcellInfo, 168
nas_PhyCaAggScellDlBw, 168
nas_PhyCaAggScellIndex, 169
nas_PhyCaAggScellIndType, 169
nas_PhyCaAggScellInfo, 172
nas_RejectReasonTlv, 180
nas_RFInfoTlv, 180
nas_SccRxInfo, 188
nas_SignalStrengthTlv, 191
nas_SLQSSignalStrengthsTlv, 193
nas_timeInfo, 200
NASBandPreferenceTlv, 213
NASEmergencyModeTlv, 214
NASGWAcqOrderPrefTlv, 214
NASLTEBandPreferenceTlv, 214
NASLteNasReleaseInfoTlv, 215
NASModePreferenceTlv, 215
NASNetSelPreferenceTlv, 215
NASOTAMessageTlv, 216
NASPhyCaAggPcellInfo, 217
NASPhyCaAggScellDlBw, 217
NASPhyCaAggScellIndex, 218
NASPhyCaAggScellIndType, 218
NASPhyCaAggScellInfo, 219
NASPRLPreferenceTlv, 220
NASRoamPreferenceTlv, 221
NASServDomainPrefTlv, 221
NASTimeInfoTlv, 223
newMTMessageTlv, 223
sMSCAddressTlv, 369
sMSEtwsMessageTlv, 370
sMSOnIMSTlv, 372
transferRouteMessageTlv, 375
wds_DataULongLongTlv, 636
wds_DataULongTlv, 637
wds_DHCPLeaseOptTlv, 637
wds_DHCPLeaseStateTlv, 637
wds_DHCPProfileIdTlv, 638
wds_IPv4AdTlv, 642
Tlvresult
pack_dms_GetCustFeaturesV2_t, 224
pack_dms_SetCustFeaturesV2_t, 227
pack_dms_SetPower_t, 228
pack_dms_SetUSBComp_t, 228
pack_dms_UIMGetICCID_t, 232
pack_fms_GetImagesPreference_t, 235
pack_fms_GetStoredImages_t, 236
pack_fms_SetImagesPreference_t, 237
pack_loc_Delete_Assist_Data_t, 238
pack_loc_EventRegister_t, 240
pack_loc_SetExtPowerState_t, 240
pack_loc_SetOperationMode_t, 241
pack_loc_SLQSLOCGetBestAvailPos_t, 241
pack_loc_Start_t, 251
pack_loc_Stop_t, 251
pack_nas_SetNetworkPreference_t, 254
pack_uim_ChangePin_t, 298
pack_uim_ReadTransparent_t, 299
pack_uim_SetPinProtection_t, 300
pack_uim_UnblockPin_t, 308
pack_uim_VerifyPin_t, 309
unpack_dms_GetBandCapability_t, 401
unpack_dms_GetCrashAction_t, 401
unpack_dms_GetCustFeature_t, 402
unpack_dms_GetCustFeaturesV2_t, 402
unpack_dms_GetDeviceCap_t, 403
unpack_dms_GetDeviceHardwareRev_t, 404
unpack_dms_GetDeviceMfr_t, 404
unpack_dms_GetDeviceSerialNumbers_t, 405
unpack_dms_GetFirmwareInfo_t, 406
unpack_dms_GetFirmwareRevision_t, 406
unpack_dms_GetFirmwareRevisions_t, 407
unpack_dms_GetFSN_t, 407
unpack_dms_GetIMSI_t, 408
unpack_dms_GetManufacturer_t, 408
unpack_dms_GetModelID_t, 409
unpack_dms_GetNetworkTime_t, 409
unpack_dms_GetOfflineReason_t, 410
unpack_dms_GetPower_t, 411
unpack_dms_GetPRLVersion_t, 411
unpack_dms_GetUSBComp_t, 412
unpack_dms_GetVoiceNumber_t, 412
unpack_dms_ResetToFactoryDefaults_t, 413
unpack_dms_SetActivationStatusCallback_t, 413
unpack_dms_SetCustFeature_t, 414
unpack_dms_SetCustFeaturesV2_t, 414
unpack_dms_SetEventReport_ind_t, 415

unpack_dms_SetEventReport_t, 415
unpack_dms_SetFirmwarePreference_t, 415
unpack_dms_SetPower_t, 415
unpack_dms_SetUSBComp_t, 416
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 416
unpack_dms_SLQSDmsSwiGetResetInfo_t, 417
unpack_dms_SLQSDmsSwiIndicationRegister_t, 418
unpack_dms_SLQSGetERIFile_t, 421
unpack_dms_SLQSSwiClearDyingGaspStatistics_t, 422
unpack_dms_SLQSSwiGetCrashInfo_t, 422
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 423
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 423
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 425
unpack_dms_SLQSSwiGetHostDevInfo_t, 426
unpack_dms_SLQSSwiGetOSInfo_t, 427
unpack_dms_SLQSSwiGetSerialNoExt_t, 427
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 428
unpack_dms_SLQSSwiSetHostDevInfo_t, 428
unpack_dms_SLQSSwiSetOSInfo_t, 429
unpack_dms_SLQSUIMGetState_t, 429
unpack_dms_UIMGetControlKeyStatus_t, 430
unpack_dms_UIMGetICCID_t, 431
unpack_dms_UIMGetPINStatus_t, 432
unpack_dms_UIMSetControlKeyProtection_t, 433
unpack_dms_UIMSetPINProtection_t, 433
unpack_dms_UIMUnblockControlKey_t, 434
unpack_fms_GetImagesPreference_t, 434
unpack_fms_GetStoredImages_t, 435
unpack_fms_SetImagesPreference_t, 436
unpack_loc_BestAvailPos_Ind_t, 441
unpack_loc_Delete_Assist_Data_t, 442
unpack_loc_DeleteAssistData_Ind_t, 443
unpack_loc_EngineState_Ind_t, 443
unpack_loc_EventRegister_t, 443
unpack_loc_GnssSvInfo_Ind_t, 444
unpack_loc_PositionRpt_Ind_t, 450
unpack_loc_SetExtPowerConfig_Ind_t, 451
unpack_loc_SetExtPowerState_t, 452
unpack_loc_SetOperationMode_Ind_t, 452
unpack_loc_SetOperationMode_t, 453
unpack_loc_SLQSLOCGetBestAvailPos_t, 453
unpack_loc_Start_t, 453
unpack_loc_Stop_t, 454
unpack_nas_GetNetworkPreference_t, 458
unpack_nas_SetNetworkPreference_t, 464
unpack_nas_SetServingSystemCallback_ind_t, 465
unpack_nas_SlqsGetLTECphyCAInfo_t, 466
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 485
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 487
unpack_uim_ChangePin_t, 540
unpack_uim_GetCardStatus_t, 540
unpack_uim_ReadTransparent_t, 541
unpack_uim_SetPinProtection_t, 542
unpack_uim_UnblockPin_t, 547
unpack_uim_VerifyPin_t, 548
unpack_wds_SLQSCreateProfile_t, 584
unpack_wds_SLQSGetProfileSettings_t, 589
unpack_wds_SLQSSetIPFamilyPreference_t, 591
toServiceId
sms_BroadcastConfig, 356
TokenBucket
unpack_qos_swiQosFlow_t, 517
tokenRate
unpack_qos_tokenBucket_t, 517
tosMask
LibPackTFTIDParams, 84
total_rx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 507
total_rx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 507
total_tx_bytes
unpack_qos_SLQSQosSwiReadDataStats_t, 507
total_tx_bytes_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 507
total_tx_pkt
unpack_qos_SLQSQosSwiReadDataStats_t, 507
total_tx_pkt_drp
unpack_qos_SLQSQosSwiReadDataStats_t, 507
TrackAreaCode
unpack_nas_SLQSGetServingSystem_t, 471
TrafficClass
unpack_qos_swiQosFlow_t, 517
trafficClass
LibPackUMTSQoS, 87
wds_UMTSMinQoS, 649
trafficPriority
LibPackUMTSQoS, 87
wds_UMTSMinQoS, 649
TranDstPort
unpack_qos_swiQosFilter_t, 513
TranSrcPort
unpack_qos_swiQosFilter_t, 513
TransCap
sms_transLayerInfo, 368
TransType
sms_transLayerInfo, 368
transactionID
sMSTransferRouteMTMessage, 373
transferDelay
LibPackUMTSQoS, 87
wds_UMTSMinQoS, 649
TransferRouteMTMessageInfo
transferRouteMessageTlv, 375
transferRouteMessageTlv, 375
TlvPresent, 375
TransferRouteMTMessageInfo, 375
transferStatInd, 375
StatsMask, 376
StatsPeriod, 376

transferStats
pack_wds_SLQSSetWdsEventCallback_t, 343
trueSrvStatus
nas_GSMSrvStatusInfo, 129
tx_bytes
unpack_QosFlowStat_t, 519
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
tx_bytes_drp
unpack_QosFlowStat_t, 519
tx_pkt
unpack_QosFlowStat_t, 519
tx_pkt_drp
unpack_QosFlowStat_t, 519
tx_pkts
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
TxDropConutTlv
unpack_RMTransferStatistics_ind_t, 520
txOKBytesCount
unpack_wds_SLQSGetDUNCallInfo_t, 588
TxOkByteCountTlv
unpack_RMTransferStatistics_ind_t, 520
TxOkConutTlv
unpack_RMTransferStatistics_ind_t, 520
txPower
nas_txInfo, 202
TxQFilter
unpack_qos_QosFlowInfo_t, 503
TxQFlowGranted
unpack_qos_QosFlowInfo_t, 503
type
pack_wds_GetDefaultProfileNum_t, 329
pack_wds_SetDefaultProfileNum_t, 332
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 416
unpack_dms_SLQSDmsSwiGetResetInfo_t, 417
unpack_qmi_t, 499
typeSize
unpack_sms_GetSMSCAddress_t, 522

u16PRLVersion
unpack_dms_GetPRLVersion_t, 411
u8PRLPreference
unpack_dms_GetPRLVersion_t, 411
UDPDstPort
unpack_qos_swiQosFilter_t, 513
UDPSrcPort
unpack_qos_swiQosFilter_t, 513
ULONG
SwiDataTypes.h, 878
ULONGLONG
SwiDataTypes.h, 878
UMTSGrantedQoS
unpack_wds_SLQSGetRuntimeSettings_t, 591
UMTSInstInfo
nas_UMTSInfo, 204
UMTSLTENbrCell
nas_WCDMAInfoLTENeighborCell, 209
UMTSReqQoS
LibPackUMTSReqQoSSigInd, 87
UNIQUE_ID_LEN
dms.h, 665
UNUSEDPARAM
common.h, 655
SwiDataTypes.h, 877
uPhyCardStatus
slot_t, 353
uPhySlotStatus
slot_t, 353
USBComp
pack_dms_SetUSBComp_t, 228
unpack_dms_GetUSBComp_t, 412
USHORT
SwiDataTypes.h, 878
USSInformation
pack_voice_SLQSVoiceOrigUSSDNoWait_t, 321
UUSData
voice_UUSInfo, 634
UUSDatalen
voice_UUSInfo, 634
UUSDcs
voice_UUSInfo, 634
UUSType
voice_UUSInfo, 634
uarfcn
nas_lteWcdmaCellInfo, 157
nas_UMTSInfo, 203
nas_wcdmaUARFCN, 213
ueInIdle
nas_LTEInfoInterfreq, 144
nas_LTEInfoIntrafreq, 146
nas_LTEInfoNeighboringGSM, 147
nas_LTEInfoNeighboringWCDMA, 148
uim.h, 889
MAX_ICCID_LENGTH, 892
MAX_NO_OF_SLOTS, 892
MAX_SLOTS_STATUS, 892
pack_uim_ChangePin, 892
pack_uim_GetCardStatus, 892
pack_uim_ReadTransparent, 893
pack_uim_SLQSUIMAuthenticate, 893
pack_uim_SLQSUIMDepersonalization, 894
pack_uim_SLQSUIMEventRegister, 894
pack_uim_SLQSUIMGetConfiguration, 894
pack_uim_SLQSUIMGetFileAttributes, 895
pack_uim_SLQSUIMGetSlotsStatus, 895
pack_uim_SLQSUIMPowerDown, 896
pack_uim_SLQSUIMPowerUp, 896
pack_uim_SLQSUIMRefreshComplete, 896
pack_uim_SLQSUIMRefreshGetLastEvent, 897
pack_uim_SLQSUIMRefreshOK, 897
pack_uim_SLQSUIMRefreshRegister, 897
pack_uim_SLQSUIMReset, 898
pack_uim_SLQSUIMSwitchSlot, 898
pack_uim_SetPinProtection, 893
pack_uim_UnblockPin, 898

pack_uim_VerifyPin, 899
unpack_uim_ChangePin, 899
unpack_uim_GetCardStatus, 900
unpack_uim_ReadTransparent, 900
unpack_uim_SLQSUIMAuthenticate, 901
unpack_uim_SLQSUIMDepersonalization, 901
unpack_uim_SLQSUIMEventRegister, 902
unpack_uim_SLQSUIMGetConfiguration, 902
unpack_uim_SLQSUIMGetFileAttributes, 902
unpack_uim_SLQSUIMGetSlotsStatus, 903
unpack_uim_SLQSUIMPowerDown, 903
unpack_uim_SLQSUIMPowerUp, 903
unpack_uim_SLQSUIMRefreshComplete, 904
unpack_uim_SLQSUIMRefreshGetLastEvent, 904
unpack_uim_SLQSUIMRefreshOK, 904
unpack_uim_SLQSUIMRefreshRegister, 905
unpack_uim_SLQSUIMReset, 905
unpack_uim_SLQSUIMSetStatusChangeCall-Back_ind, 905
unpack_uim_SLQSUIMSwitchSlot, 906
unpack_uim_SetPinProtection, 900
unpack_uim_SetUimSlotStatusChangeCallback_-ind, 901
unpack_uim_UnblockPin, 906
unpack_uim_VerifyPin, 906
uim_UIMSessionInformation, 398
aid, 399
aidLength, 399
sessionType, 399
uim_appStatus, 376
aidLength, 378
aidVal, 378
appState, 378
appType, 378
persoFeature, 378
persoRetries, 378
persoState, 378
persoUnblockRetries, 378
pin1Retries, 379
pin1State, 379
pin2Retries, 379
pin2State, 379
puk1Retries, 379
puk2Retries, 379
univPin, 379
uim_authenticateResult, 379
content, 379
contentLen, 379
uim_authenticationData, 379
context, 380
data, 380
dataLen, 380
uim_cardResult, 380
sw1, 381
sw2, 381
uim_cardStatus, 381
index1xPri, 382
index1xSec, 382
indexGwPri, 382
indexGwSec, 382
numSlot, 382
SlotInfo, 382
uim_changeUIMPIN, 382
oldPINLen, 383
oldPINVal, 383
pinID, 383
pinLen, 383
pinVal, 383
uim_depersonalizationInformation, 383
ckLen, 384
ckVal, 384
feature, 384
operation, 384
uim_encryptedPIN1, 384
pin1Len, 385
pin1Val, 385
uim_fileAttributes, 385
fileID, 388
fileSize, 388
fileType, 388
rawLen, 388
rawValue, 388
recordCount, 388
recordSize, 388
secActivate, 388
secActivateMask, 388
secDeactivate, 388
secDeactivateMask, 388
secIncrease, 388
secIncreaseMask, 388
secRead, 388
secReadMask, 388
secWrite, 388
secWriteMask, 388
uim_fileInfo, 388
fileID, 389
path, 389
pathLen, 389
uim_hotSwapStatus, 389
hotSwap, 389
hotSwapLength, 389
uim_personalizationStatus, 389
feature, 390
numFeatures, 390
unblockLeft, 390
verifyLeft, 390
uim_readResult, 390
content, 391
contentLen, 391
uim_readTransparentInfo, 391
length, 391
offset, 391
uim_refreshevent, 391
aid, 393
aidLength, 393
arrfileInfo, 393

mode, 393
numOfFiles, 393
sessionType, 393
stage, 393
uim_registerRefresh, 393
arrfileInfo, 394
numFiles, 394
registerFlag, 394
voteForInit, 394
uim_remainingRetries, 394
unblockLeft, 394
verifyLeft, 394
uim_sessionInformation, 395
aid, 395
aidLength, 395
sessionType, 395
uim_setPINProtection, 395
pinID, 396
pinLength, 396
pinOperation, 396
pinValue, 396
uim_slotInfo, 396
AppStatus, 398
cardState, 398
errorState, 398
numApp, 398
upinRetries, 398
upinState, 398
upukRetries, 398
uim_unblockUIMPIN, 399
newPINLen, 399
newPINVal, 399
pinID, 399
pukLen, 399
pukVal, 400
uim_verifyUIMPIN, 400
pinID, 400
pinLen, 400
pinVal, 400
uimSlotStatus
slots_t, 355
ulData
wds_DataULongTlv, 637
ulMask
rmTrasnferStaticsReq, 351
ulMax
_PackSwiAvmsSetSettingsPeriodInfo, 29
ulMin
_PackSwiAvmsSetSettingsPeriodInfo, 29
ulPhysicalSlot
pack_uim_SLQSUIMSwitchSlot_t, 307
ulPkgDownloadComplete
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
ulPkgDownloadSize
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
ullData
wds_DataULongLongTlv, 636
umtsEcio
nas_UMTSinstInfo, 204
umtsInst
nas_UMTSInfo, 203
umtsLTENbrCellLen
nas_WCDMAInfoLTENeighborCell, 209
umtsPsc
nas_UMTSinstInfo, 204
umtsRscp
nas_UMTSinstInfo, 204
umtsUarfcn
nas_UMTSinstInfo, 204
UnPackGetProfileSettingOut, 597
curProfile, 598
pExtErrCode, 598
unblockLeft
uim_personalizationStatus, 390
uim_remainingRetries, 394
unblockRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 430
unpack_dms_UIMSetPINProtection_t, 433
unpack_dms_UIMUnblockControlKey_t, 434
uniqueID
image_info_t, 58
univPin
appStats, 44
uim_appStatus, 379
universalTime
unpack_nas_SLQSNasNetworkTimeCallBack_ind-_t, 483
unpack_QosFlowStat_t, 518
bearerId, 519
tx_bytes, 519
tx_bytes_drp, 519
tx_pkt, 519
tx_pkt_drp, 519
unpack_RMTransferStatistics_ind_t, 519
RxDropConutTlv, 520
RxOkByteCountTlv, 520
RxOkConutTlv, 520
TxDropConutTlv, 520
TxOkByteCountTlv, 520
TxOkConutTlv, 520
unpack_dms_ActivateAutomatic
dms.h, 687
unpack_dms_GetActivationState
dms.h, 687
unpack_dms_GetActivationState_t, 400
state, 401
unpack_dms_GetBandCapability
dms.h, 688
unpack_dms_GetBandCapability_t, 401
BandCapability, 401
Tlvresult, 401
unpack_dms_GetCrashAction
dms.h, 688
unpack_dms_GetCrashAction_t, 401

DevCrashState, 401
Tlvresult, 401
unpack_dms_GetCustFeature
dms.h, 689
unpack_dms_GetCustFeature_t, 401
DHCPRelayEnabled, 402
DisableIMSI, 402
GPSLPM, 402
GPSSel, 402
GpsEnable, 402
IPFamSupport, 402
IsVoiceEnabled, 402
RMAutoConnect, 402
SMSSupport, 402
Tlvresult, 402
unpack_dms_GetCustFeaturesV2
dms.h, 689
unpack_dms_GetCustFeaturesV2_t, 402
GetCustomFeatureV2, 402
Tlvresult, 402
unpack_dms_GetDeviceCap
dms.h, 689
unpack_dms_GetDeviceCap_t, 403
DataServiceCapability, 403
MaxRXChannelRate, 403
MaxTXChannelRate, 403
RadioIfaces, 403
RadioIfacesSize, 403
SimCapability, 403
Tlvresult, 403
unpack_dms_GetDeviceCapabilities
dms.h, 689
unpack_dms_GetDeviceCapabilities_t, 403
dataServiceCaCapability, 404
maxRxChannelRate, 404
maxTxChannelRate, 404
RadioIfaces, 404
radioIfacesSize, 404
simCapability, 404
unpack_dms_GetDeviceHardwareRev
dms.h, 690
unpack_dms_GetDeviceHardwareRev_t, 404
String, 404
stringSize, 404
Tlvresult, 404
unpack_dms_GetDeviceMfr
dms.h, 690
unpack_dms_GetDeviceMfr_t, 404
String, 404
stringSize, 404
Tlvresult, 404
unpack_dms_GetDeviceSerialNumbers
dms.h, 690
unpack_dms_GetDeviceSerialNumbers_t, 404
ESNString, 405
esnSize, 405
IMEIString, 405
imeiSize, 405
imeiSvnSize, 405
ImeiSvnString, 405
MEIDString, 405
meidSize, 405
Tlvresult, 405
unpack_dms_GetFSN
dms.h, 692
unpack_dms_GetFSN_t, 407
String, 407
Tlvresult, 407
unpack_dms_GetFirmwareInfo
dms.h, 691
unpack_dms_GetFirmwareInfo_t, 405
appversion_str, 406
bootversion_str, 406
carrier_str, 406
cur_carr_name, 406
cur_carr_rev, 406
modelid_str, 406
packageid_str, 406
priversion_str, 406
sku_str, 406
Tlvresult, 406
unpack_dms_GetFirmwareRevision
dms.h, 691
unpack_dms_GetFirmwareRevision_t, 406
AMSSString, 406
amssSize, 406
PRIString, 406
Tlvresult, 406
unpack_dms_GetFirmwareRevisions
dms.h, 691
unpack_dms_GetFirmwareRevisions_t, 406
AMSSString, 407
amssSize, 407
bootSize, 407
BootString, 407
PRIString, 407
priSize, 407
Tlvresult, 407
unpack_dms_GetHardwareRevision
dms.h, 692
unpack_dms_GetHardwareRevision_t, 407
hwVer, 408
unpack_dms_GetIMSI
dms.h, 692
unpack_dms_GetIMSI_t, 408
imsi, 408
Tlvresult, 408
unpack_dms_GetManufacturer
dms.h, 693
unpack_dms_GetManufacturer_t, 408
manufacturer, 408
Tlvresult, 408
unpack_dms_GetModelID
dms.h, 693
unpack_dms_GetModelID_t, 408
modelid, 409

Tlvresult, 409
unpack_dms_GetNetworkTime
dms.h, 693
unpack_dms_GetNetworkTime_t, 409
source, 409
timestamp, 409
Tlvresult, 409
unpack_dms_GetOfflineReason
dms.h, 694
unpack_dms_GetOfflineReason_t, 410
pReasonMask, 410
pbPlatform, 410
Tlvresult, 410
unpack_dms_GetPRLVersion
dms.h, 694
unpack_dms_GetPRLVersion_t, 411
Tlvresult, 411
u16PRLVersion, 411
u8PRLPreference, 411
unpack_dms_GetPower
dms.h, 694
unpack_dms_GetPower_t, 410
HardwareControlledMode, 411
OfflineReason, 411
OperationMode, 411
Tlvresult, 411
unpack_dms_GetSerialNumbers
dms.h, 695
unpack_dms_GetSerialNumbers_t, 411
esn, 411
imei_no, 412
imeisv_svn, 412
meid, 412
unpack_dms_GetUSBComp
dms.h, 695
unpack_dms_GetUSBComp_t, 412
NumSupUSBComps, 412
SupUSBComps, 412
Tlvresult, 412
USBComp, 412
unpack_dms_GetVoiceNumber
dms.h, 695
unpack_dms_GetVoiceNumber_t, 412
MIN, 412
minSize, 412
Tlvresult, 412
VoiceNumber, 412
voiceNumberSize, 412
unpack_dms_ResetToFactoryDefaults
dms.h, 696
unpack_dms_ResetToFactoryDefaults_t, 412
Tlvresult, 413
unpack_dms_SLQSDmsSwiGetResetInfo
dms.h, 699
unpack_dms_SLQSDmsSwiGetResetInfo_Ind
dms.h, 699
unpack_dms_SLQSDmsSwiGetResetInfo_Ind_t, 416
source, 416
Tlvresult, 416
type, 416
unpack_dms_SLQSDmsSwiGetResetInfo_t, 417
source, 417
Tlvresult, 417
type, 417
unpack_dms_SLQSDmsSwiIndicationRegister
dms.h, 700
unpack_dms_SLQSDmsSwiIndicationRegister_t, 417
Tlvresult, 418
unpack_dms_SLQSGetBandCapability
dms.h, 700
unpack_dms_SLQSGetBandCapability_t, 418
bandCapability, 421
LteBandCapability, 421
TdsBandCapability, 421
unpack_dms_SLQSGetERIFile
dms.h, 701
unpack_dms_SLQSGetERIFile_t, 421
eriFile, 421
Tlvresult, 421
unpack_dms_SLQSSwiClearDyingGaspStatistics
dms.h, 701
unpack_dms_SLQSSwiClearDyingGaspStatistics_t, 421
Tlvresult, 422
unpack_dms_SLQSSwiGetCrashInfo
dms.h, 701
unpack_dms_SLQSSwiGetCrashInfo_t, 422
crashInfoParam, 422
Tlvresult, 422
unpack_dms_SLQSSwiGetDyingGaspCfg
dms.h, 702
unpack_dms_SLQSSwiGetDyingGaspCfg_t, 422
pGetDyingGaspCfg, 423
Tlvresult, 423
unpack_dms_SLQSSwiGetDyingGaspStatistics
dms.h, 702
unpack_dms_SLQSSwiGetDyingGaspStatistics_t, 423
pGetDyingGaspStatistics, 423
Tlvresult, 423
unpack_dms_SLQSSwiGetFirmwareCurr
dms.h, 702
unpack_dms_SLQSSwiGetFirmwareCurr_t, 423
carrier, 424
fwvers, 424
numEntries, 424
pCurrImgInfo, 424
pkgver, 424
priver, 424
unpack_dms_SLQSSwiGetFwUpdateStatus
dms.h, 703
unpack_dms_SLQSSwiGetFwUpdateStatus_t, 424
imgType, 425
logString, 425
refData, 425
refString, 425
ResCode, 425

Tlvresult, 425
unpack_dms_SLQSSwiGetHostDevInfo
dms.h, 703
unpack_dms_SLQSSwiGetHostDevInfo_t, 426
manString, 426
modelString, 426
plasmaIDString, 426
swVerString, 426
Tlvresult, 426
unpack_dms_SLQSSwiGetOSInfo
dms.h, 703
unpack_dms_SLQSSwiGetOSInfo_t, 426
nameString, 427
Tlvresult, 427
versionString, 427
unpack_dms_SLQSSwiGetSerialNoExt
dms.h, 704
unpack_dms_SLQSSwiGetSerialNoExt_t, 427
meidString, 427
Tlvresult, 427
unpack_dms_SLQSSwiSetDyingGaspCfg
dms.h, 704
unpack_dms_SLQSSwiSetDyingGaspCfg_t, 427
Tlvresult, 428
unpack_dms_SLQSSwiSetHostDevInfo
dms.h, 704
unpack_dms_SLQSSwiSetHostDevInfo_t, 428
Tlvresult, 428
unpack_dms_SLQSSwiSetOSInfo
dms.h, 705
unpack_dms_SLQSSwiSetOSInfo_t, 428
Tlvresult, 429
unpack_dms_SLQSUIMGetState
dms.h, 705
unpack_dms_SLQSUIMGetState_t, 429
state, 429
Tlvresult, 429
unpack_dms_SetActivationStatusCallback
dms.h, 696
unpack_dms_SetActivationStatusCallback_t, 413
Tlvresult, 413
unpack_dms_SetCrashAction
dms.h, 696
unpack_dms_SetCrashAction_t, 413
notused, 413
unpack_dms_SetCustFeature
dms.h, 697
unpack_dms_SetCustFeature_t, 414
Tlvresult, 414
unpack_dms_SetCustFeaturesV2
dms.h, 697
unpack_dms_SetCustFeaturesV2_t, 414
Tlvresult, 414
unpack_dms_SetEventReport
dms.h, 697
unpack_dms_SetEventReport_ind
dms.h, 698
unpack_dms_SetEventReport_ind_t, 414
ActivationStatusTlv, 415
OperatingModeTlv, 415
Tlvresult, 415
unpack_dms_SetEventReport_t, 415
Tlvresult, 415
unpack_dms_SetFirmwarePreference
dms.h, 698
unpack_dms_SetFirmwarePreference_t, 415
Tlvresult, 415
unpack_dms_SetPower
dms.h, 698
unpack_dms_SetPower_t, 415
Tlvresult, 415
unpack_dms_SetUSBComp
dms.h, 699
unpack_dms_SetUSBComp_t, 415
Tlvresult, 416
unpack_dms_UIMChangePIN
dms.h, 705
unpack_dms_UIMGetControlKeyStatus
dms.h, 706
unpack_dms_UIMGetControlKeyStatus_t, 429
facilityState, 430
Tlvresult, 430
unblockRetriesLeft, 430
verifyRetriesLeft, 430
unpack_dms_UIMGetICCID
dms.h, 706
unpack_dms_UIMGetICCID_t, 430
String, 430
stringSize, 431
Tlvresult, 431
unpack_dms_UIMGetPINStatus
dms.h, 706
unpack_dms_UIMGetPINStatus_t, 431
p1Status, 432
p1UnblockRetriesLeft, 432
p1VerifyRetriesLeft, 432
p2Status, 432
p2UnblockRetriesLeft, 432
p2VerifyRetriesLeft, 432
Tlvresult, 432
unpack_dms_UIMSetControlKeyProtection
dms.h, 707
unpack_dms_UIMSetControlKeyProtection_t, 432
Tlvresult, 433
verifyRetriesLeft, 433
unpack_dms_UIMSetPINProtection
dms.h, 707
unpack_dms_UIMSetPINProtection_t, 433
Tlvresult, 433
unblockRetriesLeft, 433
verifyRetriesLeft, 433
unpack_dms_UIMUnblockControlKey
dms.h, 707
unpack_dms_UIMUnblockControlKey_t, 433
Tlvresult, 434
unblockRetriesLeft, 434

unpack_dms_UIMUnblockPIN
dms.h, 708
unpack_dms_UIMVerifyPIN
dms.h, 708
unpack_dms_ValidateSPC
dms.h, 708
unpack_fms_GetImagesPreference
fms.h, 711
unpack_fms_GetImagesPreference_t, 434
ImageListSize, 434
pImageList, 434
Tlvresult, 434
unpack_fms_GetStoredImages
fms.h, 711
unpack_fms_GetStoredImages_t, 435
imageList, 435
imagelistSize, 435
Tlvresult, 435
unpack_fms_SetImagesPreference
fms.h, 711
unpack_fms_SetImagesPreference_t, 435
ImageTypes, 436
ImageTypesSize, 436
Tlvresult, 436
unpack_loc_BestAvailPos_Ind
loc.h, 728
unpack_loc_BestAvailPos_Ind_t, 436
pAltitudeWrtEllipsoid, 440
pAltitudeWrtMeanSeaLevel, 440
pGpsTime, 440
pHeading, 440
pHeadingUnc, 440
pHorCirConf, 440
pHorEllpConf, 440
pHorReliability, 440
pHorUncCircular, 441
pHorUncEllipseOrientAzimuth, 441
pHorUncEllipseSemiMajor, 441
pHorUncEllipseSemiMinor, 441
pLatitude, 441
pLongitude, 441
pMagneticDeviation, 441
pPrecisionDilution, 441
pSensorDataUsage, 441
pSpeedHorizontal, 441
pSpeedUnc, 441
pSpeedVertical, 441
pSpeedVerticalUnc, 441
pSvUsedforFix, 441
pTechnologyMask, 441
pTimeSrc, 441
pTimeUnc, 441
pTimestampUtc, 441
pVertConfidence, 441
pVertReliability, 441
pVertUnc, 441
pXid, 441
status, 441
Tlvresult, 441
unpack_loc_Delete_Assist_Data_t, 441
Tlvresult, 442
unpack_loc_DeleteAssistData
loc.h, 729
unpack_loc_DeleteAssistData_Ind
loc.h, 729
unpack_loc_DeleteAssistData_Ind_t, 442
status, 442
Tlvresult, 443
unpack_loc_EngineState_Ind
loc.h, 729
unpack_loc_EngineState_Ind_t, 443
engineState, 443
Tlvresult, 443
unpack_loc_EventRegister
loc.h, 730
unpack_loc_EventRegister_t, 443
Tlvresult, 443
unpack_loc_GnssSvInfo_Ind
loc.h, 730
unpack_loc_GnssSvInfo_Ind_t, 444
altitudeAssumed, 444
pSatelliteInfo, 444
Tlvresult, 444
unpack_loc_PositionRpt_Ind
loc.h, 730
unpack_loc_PositionRpt_Ind_t, 444
pAltitudeAssumed, 449
pAltitudeWrtEllipsoid, 449
pAltitudeWrtMeanSeaLevel, 449
pFixId, 449
pGpsTime, 449
pHeading, 449
pHeadingUnc, 450
pHorConfidence, 450
pHorReliability, 450
pHorUncCircular, 450
pHorUncEllipseOrientAzimuth, 450
pHorUncEllipseSemiMajor, 450
pHorUncEllipseSemiMinor, 450
pLatitude, 450
pLeapSeconds, 450
pLongitude, 450
pMagneticDeviation, 450
pPrecisionDilution, 450
pSensorDataUsage, 450
pSpeedHorizontal, 450
pSpeedUnc, 450
pSpeedVertical, 450
pSvUsedforFix, 450
pTechnologyMask, 450
pTimeSrc, 450
pTimeUnc, 450
pTimestampUtc, 450
pVertConfidence, 450
pVertReliability, 450
pVertUnc, 450

sessionId, 450
sessionStatus, 450
Tlvresult, 450
unpack_loc_SLQSLOCGetBestAvailPos
loc.h, 732
unpack_loc_SLQSLOCGetBestAvailPos_t, 453
Tlvresult, 453
unpack_loc_SLQSLOCInjectPosition
loc.h, 732
unpack_loc_SLQSLOCInjectSensorData
loc.h, 733
unpack_loc_SLQSLOCInjectUTCTime
loc.h, 733
unpack_loc_SLQSLOCSetCradleMountConfig
loc.h, 733
unpack_loc_SetExtPowerConfig_Ind
loc.h, 731
unpack_loc_SetExtPowerConfig_Ind_t, 451
status, 451
Tlvresult, 451
unpack_loc_SetExtPowerState
loc.h, 731
unpack_loc_SetExtPowerState_t, 451
Tlvresult, 452
unpack_loc_SetOperationMode
loc.h, 731
unpack_loc_SetOperationMode_Ind
loc.h, 732
unpack_loc_SetOperationMode_Ind_t, 452
status, 452
Tlvresult, 452
unpack_loc_SetOperationMode_t, 452
Tlvresult, 453
unpack_loc_Start
loc.h, 733
unpack_loc_Start_t, 453
Tlvresult, 453
unpack_loc_Stop
loc.h, 734
unpack_loc_Stop_t, 453
Tlvresult, 454
unpack_nas_GetACCOLC
nas.h, 759
unpack_nas_GetANAAAAuthenticationStatus
nas.h, 759
unpack_nas_GetCDMANetworkParameters
nas.h, 759
unpack_nas_GetCDMANetworkParameters_t, 454
Application, 454
Broadcast, 454
CustomSCP, 455
ForceRev0, 455
Protocol, 455
RegForeignNID, 455
RegForeignSID, 455
RegHomeSID, 455
Roaming, 455
SCI, 455
SCM, 455
unpack_nas_GetHomeNetwork
nas.h, 760
unpack_nas_GetHomeNetwork3GPP2
nas.h, 760
unpack_nas_GetHomeNetwork3GPP2_t, 455
nameSize, 456
pMCC, 456
pMNC, 456
pNID, 457
pName, 456
pNw2DescDisp, 457
pNw2DescEnc, 457
pNw2DescLen, 457
pNw2MCC, 457
pNw2MNC, 457
pNw2Name, 457
pSID, 457
unpack_nas_GetHomeNetwork_t, 457
mcc, 457
mnc, 457
name, 457
nid, 457
sid, 457
unpack_nas_GetNetworkPreference
nas.h, 760
unpack_nas_GetNetworkPreference_t, 457
ActiveTechPref, 458
Duration, 458
PersistentTechPref, 458
Tlvresult, 458
unpack_nas_GetRFInfo
nas.h, 760
unpack_nas_GetRFInfo_t, 458
instancesSize, 459
RFBandInfoElements, 459
unpack_nas_GetServingNetwork
nas.h, 761
unpack_nas_GetServingNetwork_t, 459
CSDomain, 460
DataCaps, 460
DataCapsLen, 460
MCC, 460
MNC, 460
Name, 460
nameSize, 460
PSDomain, 460
RAN, 460
RadioIfaces, 460
RadioIfacesSize, 460
RegistrationState, 460
Roaming, 460
unpack_nas_GetServingNetworkCapabilities
nas.h, 761
unpack_nas_GetServingNetworkCapabilities_t, 460
DataCaps, 460
DataCapsLen, 460
unpack_nas_GetSignalStrengths

nas.h, 761
unpack_nas_GetSignalStrengths_t, 460
len, 461
radio, 461
rssi, 461
unpack_nas_InitiateDomainAttach
nas.h, 762
unpack_nas_PerformNetworkScan
nas.h, 762
unpack_nas_PerformNetworkScan_t, 461
p3GppNetworkInfoInstances, 461
p3GppNetworkInstanceSize, 461
pPCSInstance, 461
pPCSInstanceSize, 461
pRATINstance, 461
pRATInstanceSize, 461
pScanResult, 461
unpack_nas_SLQSConfigSigInfo
nas.h, 765
unpack_nas_SLQSGetErrorRate
nas.h, 765
unpack_nas_SLQSGetErrorRate_t, 465
pCDMAFrameErrRate, 466
pGSMBER, 466
pHDRPackErrRate, 466
pWCDMABER, 466
unpack_nas_SLQSGetNetworkTime
nas.h, 765
unpack_nas_SLQSGetNetworkTime_t, 467
p3GPP2TimeInfo, 467
p3GPPTimeInfo, 467
unpack_nas_SLQSGetOperatorNameData
nas.h, 766
unpack_nas_SLQSGetOperatorNameData_t, 467
pNITZInformation, 468
pOperatorNameString, 468
pOperatorPLMNList, 468
pPLMNNetworkName, 468
pSrvcProviderName, 468
unpack_nas_SLQSGetPLMNName
nas.h, 766
unpack_nas_SLQSGetPLMNName_t, 468
longName, 468
longNameCI, 468
longNameEn, 468
longNameLen, 468
longNameSB, 468
shortName, 468
shortNameCI, 468
shortNameEn, 469
shortNameLen, 469
shortNameSB, 469
spn, 469
spnEncoding, 469
spnLength, 469
unpack_nas_SLQSGetServingSystem
nas.h, 766
unpack_nas_SLQSGetServingSystem_t, 469
BasestationID, 470
BasestationLatitude, 470
BasestationLongitude, 470
CDMASystemInfoExt, 470
CallBarStatus, 470
CellID, 470
ConcSvcInfo, 470
CurrentPLMN, 470
DTMInd, 470
DataSrvCapabilities, 470
DefaultRoamInd, 470
DetailedSvcInfo, 470
Gpp2TimeZone, 470
GppNetworkDSTAdjustment, 471
GppTimeZone, 471
HdrPersonality, 471
Lac, 471
NetworkID, 471
PRLInd, 471
RoamIndicatorVal, 471
RoamingIndicatorList, 471
ServingSystem, 471
SystemID, 471
TrackAreaCode, 471
unpack_nas_SLQSGetSignalStrength
nas.h, 767
unpack_nas_SLQSGetSignalStrength_t, 471
ecioList, 472
ecioListLen, 472
errorRateList, 472
errorRateListLen, 472
Io, 472
ltersrp, 472
ltesnr, 472
rsrqInfo, 472
rxSignalStrengthList, 472
rxSignalStrengthListLen, 472
signalStrengthReqMask, 472
sinr, 472
unpack_nas_SLQSGetSysInfo
nas.h, 767
unpack_nas_SLQSGetSysInfo_t, 472
pAddCDMASysInfo, 474
pAddGSMSysInfo, 474
pAddHDRSysInfo, 474
pAddLTESysInfo, 474
pCDMASrvStatusInfo, 474
pCDMASysInfo, 474
pGSMCallBarringSysInfo, 474
pGSMCipherDomainSysInfo, 474
pGSMSrvStatusInfo, 474
pGSMSysInfo, 474
pHDRSrvStatusInfo, 474
pHDRSysInfo, 474
pLTESrvStatusInfo, 474
pLTESysInfo, 474
pLTEVoiceSupportSysInfo, 475
pWCDMASysInfo, 475

unpack_nas_SLQSGetSysSelectionPref
nas.h, 767
unpack_nas_SLQSGetSysSelectionPref_t, 475
pBandPref, 478
pEmerMode, 478
pGWAcqOrderPref, 478
pLTEBandPref, 478
pModePref, 478
pNetSelPref, 478
pPRLPref, 478
pRoamPref, 478
pSrvDomainPref, 478
unpack_nas_SLQSInitiateNetworkRegistration
nas.h, 768
unpack_nas_SLQSNASSwiGetChannelLock
nas.h, 771
unpack_nas_SLQSNASSwiGetChannelLock_t, 484
unpack_nas_SLQSNASSwiSetChannelLock
nas.h, 772
unpack_nas_SLQSNasConfigSigInfo2
nas.h, 768
unpack_nas_SLQSNasGet3GPP2Subscription
nas.h, 768
unpack_nas_SLQSNasGet3GPP2Subscription_t, 478
pDirNum, 479
unpack_nas_SLQSNasGetCellLocationInfo
nas.h, 768
unpack_nas_SLQSNasGetCellLocationInfo_t, 479
pCDMAInfo, 480
pLTEInfoInterfreq, 480
pLTEInfoIntrafreq, 480
pUMTSInfo, 480
unpack_nas_SLQSNasGetHDRColorCode
nas.h, 769
unpack_nas_SLQSNasGetHDRColorCode_t, 481
pColorCode, 481
unpack_nas_SLQSNasGetSigInfo
nas.h, 769
unpack_nas_SLQSNasGetSigInfo_t, 481
CDMASSInfo, 481
GSMSSInfo, 481
HDRSSInfo, 481
LTESSInfo, 481
unpack_nas_SLQSNasGetTxRxInfo
nas.h, 769
unpack_nas_SLQSNasGetTxRxInfo_t, 482
pRXChain0Info, 482
pRXChain1Info, 482
pTXInfo, 482
unpack_nas_SLQSNasIndicationRegisterExt
nas.h, 770
unpack_nas_SLQSNasNetworkTimeCallBack_ind
nas.h, 770
unpack_nas_SLQSNasNetworkTimeCallBack_ind_t, 482
pDayltSavAdj, 483
pRadioInterface, 483
pTimeZone, 483
universalTime, 483
unpack_nas_SLQSNasSigInfoCallback_ind
nas.h, 770
unpack_nas_SLQSNasSigInfoCallback_ind_t, 483
pRscp, 484
unpack_nas_SLQSNasSwiIndicationRegister
nas.h, 771
unpack_nas_SLQSNasSwiModemStatus
nas.h, 771
unpack_nas_SLQSNasSwiModemStatus_t, 485
commonInfo, 485
pLTEInfo, 485
unpack_nas_SLQSNasSwiOTAMessageCallback_ind
nas.h, 772
unpack_nas_SLQSNasSwiOTAMessageCallback_ind_t, 485
Info, 485
Tlvresult, 485
unpack_nas_SLQSNasSysInfoCallback_ind
nas.h, 772
unpack_nas_SLQSNasTimerCallback_ind
nas.h, 773
unpack_nas_SLQSNasTimerCallback_ind_t, 486
t3396_apn, 486
t3396_plmn_id, 486
t3396_val, 486
unpack_nas_SLQSSetBandPreference
nas.h, 773
unpack_nas_SLQSSetSignalStrengthsCallback
nas.h, 773
unpack_nas_SLQSSetSysSelectionPref
nas.h, 773
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind
nas.h, 774
unpack_nas_SLQSSetSysSelectionPrefCallBack_ind_t, 486
Info, 486
Tlvresult, 487
unpack_nas_SLQSSwiGetHDRPersonality
nas.h, 774
unpack_nas_SLQSSwiGetHDRPersonality_t, 487
pCurrentPersonality, 487
pPersonalityListLength, 487
pProtocolSubtypeElement, 487
unpack_nas_SLQSSwiGetHDRProtSubtype
nas.h, 774
unpack_nas_SLQSSwiGetHDRProtSubtype_t, 487
pAppSubType, 488
pCurrentPrsnlty, 488
pPersonalityListLength, 488
unpack_nas_SLQSSwiGetHRPDStats
nas.h, 775
unpack_nas_SLQSSwiGetHRPDStats_t, 488
pPilotSetData, 489
pUATI, 489
unpack_nas_SLQSSwiGetLteCQI
nas.h, 775
unpack_nas_SLQSSwiGetLteCQI_t, 489

ValidityCW0, 489
ValidityCW1, 489
unpack_nas_SLQSSwiGetLteSccRxInfo
nas.h, 775
unpack_nas_SLQSSwiGetLteSccRxInfo_t, 489
pSccRxInfo, 490
unpack_nas_SLQSSwiNetworkDebug
nas.h, 776
unpack_nas_SLQSSwiNetworkDebug_t, 490
pDataStatusDetail, 490
pDeviceConfigDetail, 490
pNetworkStat1x, 490
pNetworkStatEVDO, 490
pObjectVer, 490
unpack_nas_SLQSSwiPSDetach
nas.h, 776
unpack_nas_SLQSSysInfoCallback_ind_t, 491
pGSMSysInfo, 493
pHDRSysInfo, 493
pLTESysInfo, 493
pSysInfoNoChange, 493
unpack_nas_SetACCOLC
nas.h, 762
unpack_nas_SetCDMANetworkParameters
nas.h, 763
unpack_nas_SetDataCapabilitiesCallback_ind
nas.h, 763
unpack_nas_SetDataCapabilitiesCallback_ind_t, 462
dataCaps, 462
dataCapsSize, 462
unpack_nas_SetEventReportInd
nas.h, 763
unpack_nas_SetEventReportInd_t, 462
RFTlv, 462
RRTlv, 462
SLQSSSTlv, 462
SSTlv, 462
unpack_nas_SetLURejectCallback
nas.h, 763
unpack_nas_SetNasLTECphyCaIndCallback_ind
nas.h, 764
unpack_nas_SetNasLTECphyCaIndCallback_ind_t, 462
unpack_nas_SetNetworkPreference
nas.h, 764
unpack_nas_SetNetworkPreference_t, 463
Tlvresult, 464
unpack_nas_SetRFInfoCallback
nas.h, 764
unpack_nas_SetRoamingIndicatorCallback_ind
nas.h, 764
unpack_nas_SetRoamingIndicatorCallback_ind_t, 464
roaming, 465
unpack_nas_SetServingSystemCallback_ind
nas.h, 765
unpack_nas_SetServingSystemCallback_ind_t, 465
SSInfo, 465
Tlvresult, 465
unpack_nas_SlqsGetLTECphyCAInfo
nas.h, 765
unpack_nas_SlqsGetLTECphyCAInfo_t, 466
LTECphyCaInfo, 466
Tlvresult, 466
unpack_omaDmConfigTlv_t, 493
alertmsg, 494
alertmsglength, 494
state, 494
userInputReq, 494
userInputTimeout, 494
unpack_omaDmFotaTlv_t, 494
description, 496
descriptionlength, 496
fwdloadsize, 496
fwloadComplete, 496
namelength, 496
package_name, 496
sessionType, 496
severity, 496
state, 496
updateCompleteStatus, 496
userInputReq, 496
userInputTimeout, 496
version, 496
versionlength, 496
unpack_omaDmNotificationsTlv_t, 496
notification, 497
sessionStatus, 497
unpack_pds_GetPDSDefaults
pds.h, 779
unpack_pds_GetPDSDefaults_t, 497
pAccuracy, 497
pInterval, 497
pOperation, 497
pTimeout, 497
unpack_pds_GetPDSState
pds.h, 779
unpack_pds_GetPDSState_t, 497
pEnabledStatus, 498
pTrackingStatus, 498
unpack_pds_GetPortAutomaticTracking
pds.h, 780
unpack_pds_GetPortAutomaticTracking_t, 498
pbAuto, 498
unpack_pds_SetPDSDefaults
pds.h, 780
unpack_pds_SetPDSState
pds.h, 780
unpack_pds_SetPortAutomaticTracking
pds.h, 781
unpack_qmi_t, 498
msgid, 499
type, 499
xid, 499
unpack_qos_IPv4Addr_t, 499
addr, 500
subnetMask, 500
unpack_qos_IPv6Addr_t, 500

addr, 500
prefixLen, 500
unpack_qos_IPv6TrafCls_t, 500
mask, 501
val, 501
unpack_qos_Port_t, 501
port, 502
range, 502
unpack_qos_QosFlowInfo_t, 502
BearerID, 503
is_RxQFlowGranted_Available, 503
is_TxQFlowGranted_Available, 503
NumRxFilters, 503
NumTxFilters, 503
QFlowState, 503
RxQFilter, 503
RxQFlowGranted, 503
TxQFilter, 503
TxQFlowGranted, 503
unpack_qos_QosFlowInfoState_t, 503
id, 504
isNewFlow, 504
state, 504
unpack_qos_SLQSQosGetNetworkStatus
qos.h, 829
unpack_qos_SLQSQosGetNetworkStatus_t, 504
NWQoSStatus, 504
unpack_qos_SLQSQosSwiReadApnExtraParams
qos.h, 830
unpack_qos_SLQSQosSwiReadApnExtraParams_t, 504
ambr_dl, 505
ambr_dl_ext, 505
ambr_dl_ext2, 505
ambr_ul, 505
ambr_ul_ext, 505
ambr_ul_ext2, 505
apnId, 505
unpack_qos_SLQSQosSwiReadDataStats
qos.h, 830
unpack_qos_SLQSQosSwiReadDataStats_t, 506
apnId, 507
numQosFlow, 507
qosFlow, 507
total_rx_bytes, 507
total_rx_pkt, 507
total_tx_bytes, 507
total_tx_pkt, 507
unpack_qos_SLQSSetQosEventCallback
qos.h, 831
unpack_qos_SLQSSetQosEventCallback_ind
qos.h, 831
unpack_qos_SLQSSetQosEventCallback_ind_t, 507
NumFlows, 507
QosFlowInfo, 507
unpack_qos_SLQSSetQosNWStatusCallback_ind
qos.h, 831
unpack_qos_SLQSSetQosNWStatusCallback_ind_t, 507
status, 508
unpack_qos_SLQSSetQosPriEventCallback_ind
qos.h, 832
unpack_qos_SLQSSetQosPriEventCallback_ind_t, 508
event, 508
unpack_qos_SLQSSetQosStatusCallback_ind
qos.h, 833
unpack_qos_SLQSSetQosStatusCallback_ind_t, 508
event, 509
id, 509
reason, 509
status, 509
unpack_qos_Tos_t, 518
mask, 518
val, 518
unpack_qos_dataRate_t, 499
dataRateMax, 499
guaranteedRate, 499
unpack_qos_pktErrRate_t, 501
exponent, 501
multiplier, 501
unpack_qos_swiQosFilter_t, 510
EspSpi, 512
IPv4DstAddr, 512
IPv4SrcAddr, 512
IPv4Tos, 512
IPv6DstAddr, 512
IPv6Label, 512
IPv6SrcAddr, 512
IPv6TrafCls, 512
Id, 512
index, 512
is_EspSpi_Available, 512
is_IPv4DstAddr_Available, 512
is_IPv4SrcAddr_Available, 512
is_IPv4Tos_Available, 512
is_IPv6DstAddr_Available, 512
is_IPv6Label_Available, 512
is_IPv6SrcAddr_Available, 512
is_IPv6TrafCls_Available, 512
is_Id_Available, 512
is_NxtHdrProto_Available, 512
is_Precedence_Available, 512
is_TCPDstPort_Available, 512
is_TCPSrcPort_Available, 512
is_TranDstPort_Available, 513
is_TranSrcPort_Available, 513
is_UDPDstPort_Available, 513
is_UDPSrcPort_Available, 513
NxtHdrProto, 513
Precedence, 513
TCPDstPort, 513
TCPSrcPort, 513
TranDstPort, 513
TranSrcPort, 513
UDPDstPort, 513

UDPSrcPort, 513
version, 513
unpack_qos_swiQosFlow_t, 513
DataRate, 516
index, 516
is_DataRate_Available, 516
is_Jitter_Available, 516
is_Latency_Available, 516
is_LteQci_Available, 516
is_MaxAllowedPktSz_Available, 516
is_MinPolicedPktSz_Available, 516
is_PktErrRate_Available, 516
is_ProfileId3GPP2_Available, 516
is_TokenBucket_Available, 516
is_TrafficClass_Available, 516
is_val_3GPP2Pri_Available, 516
is_val_3GPPImCn_Available, 516
is_val_3GPPSigInd_Available, 516
Jitter, 516
Latency, 516
LteQci, 517
MaxAllowedPktSz, 517
MinPolicedPktSz, 517
PktErrRate, 517
ProfileId3GPP2, 517
TokenBucket, 517
TrafficClass, 517
val_3GPP2Pri, 517
val_3GPPImCn, 517
val_3GPPResResidualBER, 517
val_3GPPSigInd, 517
val_3GPPTraHdlPri, 517
unpack_qos_tokenBucket_t, 517
bucketSz, 517
peakRate, 517
tokenRate, 517
unpack_result_code_only
common.h, 658
unpack_rms_GetSMSWake
rms.h, 834
unpack_rms_GetSMSWake_t, 519
enabled, 519
wake_mask, 519
unpack_rms_SetSMSWake
rms.h, 835
unpack_sar_SLQSGetRfSarState
sar.h, 836
unpack_sar_SLQSGetRfSarState_t, 520
pSarRFState, 521
unpack_sar_SLQSSetRfSarState
sar.h, 836
unpack_sms_GetSMSCAddress
sms.h, 851
unpack_sms_GetSMSCAddress_t, 521
addressSize, 521
pSMSCAddress, 521
pSMSCType, 522
typeSize, 522
unpack_sms_SLQSDeleteSMS
sms.h, 853
unpack_sms_SLQSDeleteSMS_t, 524
unpack_sms_SLQSGetIndicationRegister
sms.h, 853
unpack_sms_SLQSGetIndicationRegister_t, 524
pGetIndicationRegInfo, 524
unpack_sms_SLQSGetMessageWaiting
sms.h, 853
unpack_sms_SLQSGetMessageWaiting_t, 524
pGetMsgWaitingInfoResp, 524
unpack_sms_SLQSGetSMS
sms.h, 854
unpack_sms_SLQSGetSMS_t, 525
message, 525
messageFormat, 525
messageSize, 525
messageTag, 525
unpack_sms_SLQSGetSMSList
sms.h, 854
unpack_sms_SLQSGetSMSList_t, 526
messageList, 526
messageListSize, 526
unpack_sms_SLQSGetSmsBroadcastConfig
sms.h, 854
unpack_sms_SLQSGetSmsBroadcastConfig_t, 525
pBroadcastConfig, 526
pCDMABroadcastConfig, 526
unpack_sms_SLQSGetTransLayerInfo
sms.h, 855
unpack_sms_SLQSGetTransLayerInfo_t, 526
pGetTransLayerInfo, 527
unpack_sms_SLQSGetTransNWRegInfo
sms.h, 855
unpack_sms_SLQSGetTransNWRegInfo_t, 527
unpack_sms_SLQSModifySMSStatus
sms.h, 855
unpack_sms_SLQSModifySMSStatus_t, 527
unpack_sms_SLQSSendAsyncSMS
sms.h, 856
unpack_sms_SLQSSetIndicationRegister
sms.h, 856
unpack_sms_SLQSSetSmsBroadcastActivation
sms.h, 856
unpack_sms_SLQSSetSmsBroadcastConfig
sms.h, 857
unpack_sms_SLQSSetSmsStorage
sms.h, 857
unpack_sms_SLQSSmsGetMaxStorageSize
sms.h, 857
unpack_sms_SLQSSmsGetMaxStorageSize_t, 527
pMaxStorageSizeResp, 528
unpack_sms_SLQSSmsGetMessageProtocol
sms.h, 858
unpack_sms_SLQSSmsGetMessageProtocol_t, 528
pMessageProtocol, 528
unpack_sms_SLQSSmsSetRoutes
sms.h, 858

unpack_sms_SLQSSwiGetSMSStorage
sms.h, 858
unpack_sms_SLQSSwiGetSMSStorage_t, 528
pSmsStorage, 529
unpack_sms_SLQSWmsMemoryFullCallBack_ind
sms.h, 859
unpack_sms_SLQSWmsMemoryFullCallBack_ind_t, 529
messageMode, 529
storageType, 529
unpack_sms_SaveSMS
sms.h, 851
unpack_sms_SaveSMS_t, 522
pMessageIndex, 522
unpack_sms_SendSMS
sms.h, 851
unpack_sms_SendSMS_t, 522
messageFailureCode, 522
messageID, 522
unpack_sms_SetNewSMSCallback
sms.h, 852
unpack_sms_SetNewSMSCallback_ind
sms.h, 852
unpack_sms_SetNewSMSCallback_ind_t, 522
ETWSTlv, 523
IMSTlv, 523
MMTlv, 523
NewMMTlv, 523
SMSCTlv, 523
TRMessageTlv, 524
unpack_sms_SetNewSMSCallback_t, 524
unpack_sms_SetSMSCAddress
sms.h, 852
unpack_swiavms_SLQSAvMSEventReportInd
swiavms.h, 873
unpack_swiavms_SLQSAvMSEventReportInd_t
swiavms.h, 862
unpack_swiavms_SLQSAvMSGetSettings
swiavms.h, 873
unpack_swiavms_SLQSAvMSGetSettings_t, 529
AutoConnect, 530
AutoReboot, 530
FwAutoSDM, 530
FwAutoUpdate, 530
FwAutodownload, 530
OMADMEnabled, 530
pAPNInfo, 530
pConnectionRetryTimers, 530
pNotificationStore, 530
pPeroidsInfo, 530
pPollingTimer, 530
resultcode, 530
unpack_swiavms_SLQSAvMSSendSelection
swiavms.h, 874
unpack_swiavms_SLQSAvMSSendSelection_t, 530
resultcode, 530
unpack_swiavms_SLQSAvMSSessionGetInfo
swiavms.h, 874
unpack_swiavms_SLQSAvMSSessionGetInfo_t
swiavms.h, 863
unpack_swiavms_SLQSAvMSSetSettings
swiavms.h, 875
unpack_swiavms_SLQSAvMSSetSettings_t, 531
resultcode, 531
unpack_swiavms_SLQSAvMSStartSession
swiavms.h, 876
unpack_swiavms_SLQSAvMSStartSession_t, 531
resultcode, 531
sessionResponse, 532
unpack_swiavms_SLQSAvMSStopSession
swiavms.h, 876
unpack_swiavms_SLQSAvMSStopSession_t, 532
resultcode, 532
unpack_swiavms_SLQSAvmsSetEventReport
swiavms.h, 875
unpack_swiavms_SLQSAvmsSetEventReport_t, 530
resultcode, 531
unpack_swiloc_SwiLocGetAutoStart
swiloc.h, 879
unpack_swiloc_SwiLocGetAutoStart_t, 532
fix_rate, 533
fix_rate_reported, 533
fix_type, 533
fix_type_reported, 533
function, 533
function_reported, 533
max_dist, 533
max_dist_reported, 533
max_time, 534
max_time_reported, 534
unpack_swiloc_SwiLocSetAutoStart
swiloc.h, 879
unpack_swioma_SLQSOMADMAlertCallback
swioma.h, 885
unpack_swioma_SLQSOMADMAlertCallback_ind
swioma.h, 886
unpack_swioma_SLQSOMADMAlertCallback_ind_t, 534
eventType, 534
unpack_swioma_SLQSOMADMCancelSession
swioma.h, 886
unpack_swioma_SLQSOMADMGetSessionInfo
swioma.h, 886
unpack_swioma_SLQSOMADMGetSessionInfo_t, 534
Date, 536
DateLength, 536
PkgDescLength, 536
PkgDescription, 536
PkgName, 537
PkgNameLength, 537
RetryCount, 537
SessionState, 537
SessionType, 537
Severity, 537
Source, 537
SourceLength, 537

Status, 537
Time, 537
TimeLength, 537
UpdateCompleteStatus, 537
unpack_swioma_SLQSOMADMGetSettings
swioma.h, 887
unpack_swioma_SLQSOMADMGetSettings_t, 537
Autosdm, 538
FOTAdownload, 538
FwAutoCheck, 538
unpack_swioma_SLQSOMADMSendSelection
swioma.h, 887
unpack_swioma_SLQSOMADMSetSettings
swioma.h, 888
unpack_swioma_SLQSOMADMStartSession
swioma.h, 888
unpack_swioma_SLQSOMADMStartSession_t, 538
FwAvailability, 539
unpack_uim_ChangePin
uim.h, 899
unpack_uim_ChangePin_t, 539
pEncryptedPIN1, 539
pIndicationToken, 539
pRemainingRetries, 540
Tlvresult, 540
unpack_uim_GetCardStatus
uim.h, 900
unpack_uim_GetCardStatus_t, 540
pCardStatus, 540
pHotSwapStatus, 540
Tlvresult, 540
unpack_uim_ReadTransparent
uim.h, 900
unpack_uim_ReadTransparent_t, 540
pCardResult, 541
pEncryptedData, 541
pIndicationToken, 541
pReadResult, 541
Tlvresult, 541
unpack_uim_SLQSUIMAuthenticate
uim.h, 901
unpack_uim_SLQSUIMAuthenticate_t, 542
pAuthenticateResult, 543
pCardResult, 543
pIndicationToken, 543
unpack_uim_SLQSUIMDepersonalization
uim.h, 901
unpack_uim_SLQSUIMDepersonalization_t, 543
pRemainingRetries, 543
unpack_uim_SLQSUIMEventRegister
uim.h, 902
unpack_uim_SLQSUIMEventRegister_t, 544
eventMask, 544
unpack_uim_SLQSUIMGetConfiguration
uim.h, 902
unpack_uim_SLQSUIMGetConfiguration_t, 544
pAutoSelection, 545
pHaltSubscription, 545
pPersonalizationStatus, 545
unpack_uim_SLQSUIMGetFileAttributes
uim.h, 902
unpack_uim_SLQSUIMGetFileAttributes_t, 545
pCardResult, 545
pFileAttributes, 545
pIndicationToken, 545
unpack_uim_SLQSUIMGetSlotsStatus
uim.h, 903
unpack_uim_SLQSUIMGetSlotsStatus_t, 545
pNumberOfPhySlot, 546
pUimSlotsStatus, 546
unpack_uim_SLQSUIMPowerDown
uim.h, 903
unpack_uim_SLQSUIMPowerUp
uim.h, 903
unpack_uim_SLQSUIMRefreshComplete
uim.h, 904
unpack_uim_SLQSUIMRefreshGetLastEvent
uim.h, 904
unpack_uim_SLQSUIMRefreshGetLastEvent_t, 546
pRefreshEvent, 546
unpack_uim_SLQSUIMRefreshOK
uim.h, 904
unpack_uim_SLQSUIMRefreshRegister
uim.h, 905
unpack_uim_SLQSUIMReset
uim.h, 905
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind
uim.h, 905
unpack_uim_SLQSUIMSetStatusChangeCallBack_ind_t, 546
unpack_uim_SLQSUIMSwitchSlot
uim.h, 906
unpack_uim_SetPinProtection
uim.h, 900
unpack_uim_SetPinProtection_t, 541
pEncryptedPIN1, 542
pIndicationToken, 542
pRemainingRetries, 542
Tlvresult, 542
unpack_uim_SetUimSlotStatusChangeCallback_ind
uim.h, 901
unpack_uim_SetUimSlotStatusChangeCallback_ind_t, 542
bNumberOfPhySlots, 542
slotsstatusChange, 542
unpack_uim_UnblockPin
uim.h, 906
unpack_uim_UnblockPin_t, 547
pEncryptedPIN1, 547
pIndicationToken, 547
pRemainingRetries, 547
Tlvresult, 547
unpack_uim_VerifyPin
uim.h, 906
unpack_uim_VerifyPin_t, 547
pEncryptedPIN1, 548

pIndicationToken, 548
pRemainingRetries, 548
Tlvresult, 548
unpack_voice_AnswerUSSD
voice.h, 923
unpack_voice_CancelUSSD
voice.h, 923
unpack_voice_OriginateUSSD
voice.h, 923
unpack_voice_SLQSOriginateUSSD
voice.h, 924
unpack_voice_SLQSOriginateUSSD_t, 548
pAlphaIDInfo, 548
pCCSuppsType, 548
pCallId, 548
pCcResultType, 548
pUSSDInfo, 548
pfailureCause, 548
unpack_voice_SLQSVoiceALSSelectLine
voice.h, 924
unpack_voice_SLQSVoiceALSSetLineSwitching
voice.h, 924
unpack_voice_SLQSVoiceAnswerCall
voice.h, 925
unpack_voice_SLQSVoiceAnswerCall_t, 549
pCallId, 549
unpack_voice_SLQSVoiceBindSubscription
voice.h, 925
unpack_voice_SLQSVoiceBurstDTMF
voice.h, 925
unpack_voice_SLQSVoiceBurstDTMF_t, 549
pCallID, 549
unpack_voice_SLQSVoiceDialCall
voice.h, 926
unpack_voice_SLQSVoiceDialCall_t, 549
pAlphaIDInfo, 550
pCCResultType, 550
pCCSUPSType, 550
pCallID, 550
unpack_voice_SLQSVoiceEndCall
voice.h, 926
unpack_voice_SLQSVoiceEndCall_t, 550
pCallId, 550
unpack_voice_SLQSVoiceGetAllCallInfo
voice.h, 926
unpack_voice_SLQSVoiceGetAllCallInfo_t, 551
pArrAlertingPattern, 553
pArrAlertingType, 553
pArrAlphaID, 553
pArrCallEndReason, 553
pArrCallInfo, 553
pArrCalledPartyNum, 553
pArrConnectPartyNum, 553
pArrDiagInfo, 553
pArrRedirPartyNum, 553
pArrRemotePartyName, 553
pArrRemotePartyNum, 553
pArrSvcOption, 553
pArrUUSInfo, 553
pVoicePrivacy, 553
unpack_voice_SLQSVoiceGetCLIP
voice.h, 928
unpack_voice_SLQSVoiceGetCLIP_t, 560
pAlphaIDInfo, 561
pCCResType, 561
pCallID, 561
pFailCause, 561
unpack_voice_SLQSVoiceGetCLIR
voice.h, 929
unpack_voice_SLQSVoiceGetCLIR_t, 561
pAlphaIDInfo, 562
pCCResType, 562
pCallID, 562
pFailCause, 562
unpack_voice_SLQSVoiceGetCNAP
voice.h, 929
unpack_voice_SLQSVoiceGetCNAP_t, 562
pAlphaIDInfo, 563
pCCResType, 563
pCallID, 563
pFailCause, 563
unpack_voice_SLQSVoiceGetCOLP
voice.h, 929
unpack_voice_SLQSVoiceGetCOLP_t, 563
pAlphaIDInfo, 564
pCCResType, 565
pCallID, 564
pFailCause, 565
unpack_voice_SLQSVoiceGetCOLR
voice.h, 930
unpack_voice_SLQSVoiceGetCOLR_t, 565
pAlphaIDInfo, 566
pCCResType, 566
pCallID, 566
pFailCause, 566
unpack_voice_SLQSVoiceGetCallBarring
voice.h, 927
unpack_voice_SLQSVoiceGetCallBarring_t, 553
pAlphaIDInfo, 554
pCCResType, 554
pCallID, 554
pFailCause, 554
pSvcClass, 554
unpack_voice_SLQSVoiceGetCallForwardingStatus
voice.h, 927
unpack_voice_SLQSVoiceGetCallForwardingStatus_t, 555
pAlphaIDInfo, 556
pCCResType, 556
pCallID, 556
pFailCause, 556
unpack_voice_SLQSVoiceGetCallInfo
voice.h, 928
unpack_voice_SLQSVoiceGetCallInfo_t, 556
pAlertType, 558
pAlertingPattern, 558

pAlphaIDInfo, 558
pCallInfo, 558
pConnectNumInfo, 558
pDiagInfo, 558
pOTASPStatus, 558
pRemotePartyName, 558
pRemotePartyNum, 558
pSrvOpt, 558
pUUSInfo, 558
pVoicePrivacy, 558
unpack_voice_SLQSVoiceGetCallWaiting
voice.h, 928
unpack_voice_SLQSVoiceGetCallWaiting_t, 558
pAlphaIDInfo, 559
pCCResType, 560
pCallID, 560
pFailCause, 560
pSvcClass, 560
unpack_voice_SLQSVoiceGetConfig
voice.h, 930
unpack_voice_SLQSVoiceGetConfig_t, 566
pAirTimerCnt, 567
pAutoAnswerStat, 568
pCurAMRConfig, 568
pCurPrefVoiceSO, 568
pCurVoiceDomainPref, 568
pCurVoicePrivacyPref, 568
pCurrTTYMode, 568
pRoamTimerCnt, 568
unpack_voice_SLQSVoiceIndicationRegister
voice.h, 930
unpack_voice_SLQSVoiceManageCalls
voice.h, 931
unpack_voice_SLQSVoiceManageCalls_t, 568
pFailCause, 568
unpack_voice_SLQSVoiceOrigUSSDNoWait
voice.h, 931
unpack_voice_SLQSVoiceSendFlash
voice.h, 931
unpack_voice_SLQSVoiceSendFlash_t, 568
pCallID, 569
unpack_voice_SLQSVoiceSetCallBarringPassword
voice.h, 932
unpack_voice_SLQSVoiceSetCallBarringPassword_t, 569
pAlphaIDInfo, 569
pCCResType, 570
pCallID, 570
pFailCause, 570
unpack_voice_SLQSVoiceSetConfig
voice.h, 932
unpack_voice_SLQSVoiceSetConfig_t, 570
pAirTimerStatus, 571
pAutoAnsStatus, 571
pPrefVoiceSOStatus, 571
pRoamTimerStatus, 571
pTTYConfigStatus, 571
pVoiceDomainPrefStatus, 571
unpack_voice_SLQSVoiceSetPreferredPrivacy
voice.h, 932
unpack_voice_SLQSVoiceSetSUPSService
voice.h, 933
unpack_voice_SLQSVoiceSetSUPSService_t, 571
pCallID, 572
pFailCause, 572
unpack_voice_SLQSVoiceStartContDTMF
voice.h, 933
unpack_voice_SLQSVoiceStartContDTMF_t, 572
pCallID, 573
unpack_voice_SLQSVoiceStopContDTMF
voice.h, 933
unpack_voice_SLQSVoiceStopContDTMF_t, 573
callID, 573
unpack_wds_DHCPv4ClientLease_ind
wds.h, 959
unpack_wds_DHCPv4ClientLease_ind_t, 573
DHCPv4LeaseOptTlv, 574
DHCPv4LeaseStateTlv, 574
IPv4AddrTlv, 574
ProfileIdTlv, 574
unpack_wds_DHCPv4ClientLeaseChange
wds.h, 959
unpack_wds_GetAutoconnect
wds.h, 959
unpack_wds_GetAutoconnect_t, 574
psetting, 574
unpack_wds_GetByteTotals
wds.h, 960
unpack_wds_GetByteTotals_t, 574
pRXTotalBytes, 574
pTXTotalBytes, 575
unpack_wds_GetConnectionRate
wds.h, 960
unpack_wds_GetConnectionRate_t, 575
currentChannelRXRate, 575
currentChannelTXRate, 575
maxChannelRXRate, 575
maxChannelTXRate, 575
unpack_wds_GetDataBearerTechnology
wds.h, 960
unpack_wds_GetDataBearerTechnology_t, 575
pDataBearer, 576
unpack_wds_GetDefaultProfile
wds.h, 961
unpack_wds_GetDefaultProfile_t, 576
apnname, 577
apnsize, 577
auth, 577
ipaddr, 577
ipaddrv6, 577
name, 577
namesize, 577
pdptype, 577
pridns, 577
pridnsv6, 577
secdns, 577

secdnsv6, 577
username, 577
usersize, 577
unpack_wds_GetDefaultProfileNum
wds.h, 961
unpack_wds_GetDefaultProfileNum_t, 577
index, 578
unpack_wds_GetDormancyState
wds.h, 961
unpack_wds_GetDormancyState_t, 578
dormancyState, 578
unpack_wds_GetLastMobileIPError
wds.h, 962
unpack_wds_GetLastMobileIPError_t, 578
error, 578
unpack_wds_GetMobileIP
wds.h, 962
unpack_wds_GetMobileIP_t, 578
mipMode, 579
unpack_wds_GetMobileIPProfile
wds.h, 962
unpack_wds_GetMobileIPProfile_t, 579
AAASPI, 579
AAAState, 579
address, 579
enabled, 579
HASPI, 579
HAState, 579
NAI, 579
naiSize, 579
primaryHA, 579
revTunneling, 580
secondaryHA, 580
unpack_wds_GetPacketStatistics
wds.h, 963
unpack_wds_GetPacketStatistics_t, 580
pRXDroppedCount, 581
pRXOKBytesLastCall, 581
pRXOkBytesCount, 581
pRXPacketErrors, 581
pRXPacketOverflows, 581
pRXPacketSuccesses, 581
pTXDroppedCount, 581
pTXOKBytesLastCall, 581
pTXOkBytesCount, 581
pTXPacketErrors, 581
pTXPacketOverflows, 581
pTXPacketSuccesses, 581
unpack_wds_GetPacketStatus
wds.h, 963
unpack_wds_GetPacketStatus_t, 581
rXDroppedCount, 582
rXOKBytesLastCall, 582
rXOkBytesCount, 582
rXPacketErrors, 582
rXPacketOverflows, 582
rXPacketSuccesses, 582
tXDroppedCount, 582
tXOKBytesLastCall, 582
tXOkBytesCount, 582
tXPacketErrors, 582
tXPacketOverflows, 582
tXPacketSuccesses, 582
unpack_wds_GetSessionDuration
wds.h, 963
unpack_wds_GetSessionDuration_t, 582
callDuration, 583
unpack_wds_GetSessionState
wds.h, 964
unpack_wds_GetSessionState_t, 583
connectionStatus, 583
unpack_wds_RMSetTransferStatistics
wds.h, 964
unpack_wds_RMSetTransferStatistics_t, 583
unpack_wds_RMTransferStatistics_ind
wds.h, 964
unpack_wds_SLQSCreateProfile
wds.h, 967
unpack_wds_SLQSCreateProfile_t, 583
pCreateProfileOut, 583
pProfileID, 583
Tlvresult, 584
unpack_wds_SLQSDeleteProfile
wds.h, 967
unpack_wds_SLQSDeleteProfile_t, 584
extendedErrorCode, 584
unpack_wds_SLQSGet3GPPConfigItem
wds.h, 967
unpack_wds_SLQSGet3GPPConfigItem_t, 584
_3gppRelease, 585
defaultPDNEnabled, 585
LTEAttachProfile, 585
profileList, 585
unpack_wds_SLQSGetCurrDataSystemStat
wds.h, 968
unpack_wds_SLQSGetCurrDataSystemStat_t, 585
currNetworkInfo, 586
networkInfoLen, 586
prefNetwork, 586
unpack_wds_SLQSGetCurrentChannelRate
wds.h, 968
unpack_wds_SLQSGetCurrentChannelRate_t, 586
max_channel_rx_rate, 587
max_channel_tx_rate, 587
unpack_wds_SLQSGetDUNCallInfo
wds.h, 969
unpack_wds_SLQSGetDUNCallInfo_t, 587
callEndReason, 588
channelRate, 588
connectionStatus, 588
dataBearerTech, 588
dormancyStatus, 588
lastCallDataBearerTech, 588
mdmCallDurationActive, 588
rxOKBytesCount, 588
txOKBytesCount, 588

unpack_wds_SLQSGetDataBearerTechnology
wds.h, 968
unpack_wds_SLQSGetDataBearerTechnology_t, 587
curDataBearerTechnology, 587
dataBearerMask, 587
lastCallDataBearerTechnology, 587
unpack_wds_SLQSGetProfileSettings
wds.h, 969
unpack_wds_SLQSGetProfileSettings_t, 588
pProfileSettings, 588
ProfileType, 589
Tlvresult, 589
unpack_wds_SLQSGetRuntimeSettings
wds.h, 969
unpack_wds_SLQSGetRuntimeSettings_t, 589
APNName, 590
Authentication, 590
DomainList, 590
GPRSGrantedQoS, 590
GWAddressV4, 590
IMCNflag, 590
IPFamilyPreference, 590
IPV6AddrInfo, 590
IPV6GWAddrInfo, 590
IPv4, 590
Mtu, 590
PDPType, 590
PrimaryDNSV4, 590
PrimaryDNSV6, 590
ProfileID, 590
ProfileName, 590
SecondaryDNSV4, 590
SecondaryDNSV6, 591
ServerAddrList, 591
SubnetMaskV4, 591
Technology, 591
UMTSGrantedQoS, 591
Username, 591
unpack_wds_SLQSModifyProfile
wds.h, 970
unpack_wds_SLQSModifyProfile_t, 591
pExtErrorCode, 591
unpack_wds_SLQSResetPacketStatics
wds.h, 970
unpack_wds_SLQSSGetDHCPv4ClientConfig
wds.h, 972
unpack_wds_SLQSSGetDHCPv4ClientConfig_t, 594
pHwConfig, 594
unpack_wds_SLQSSGetLoopback
wds.h, 972
unpack_wds_SLQSSGetLoopback_t, 595
ByteLoopbackMode, 595
ByteLoopbackMultiplier, 595
unpack_wds_SLQSSSetDHCPv4ClientConfig
wds.h, 973
unpack_wds_SLQSSSetLoopback
wds.h, 973
unpack_wds_SLQSSet3GPPConfigItem
wds.h, 970
unpack_wds_SLQSSetIPFamilyPreference
wds.h, 970
unpack_wds_SLQSSetIPFamilyPreference_t, 591
Tlvresult, 591
unpack_wds_SLQSSetPacketSrvStatusCallback
wds.h, 971
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 591
bearerID, 592
conn_status, 592
ipFamily, 592
reconfigReqd, 592
sessionEndReason, 592
techName, 592
verboseSessnEndReason, 592
verboseSessnEndReasonType, 592
unpack_wds_SLQSSetWdsEventCallback
wds.h, 971
unpack_wds_SLQSSetWdsEventCallback_ind
wds.h, 971
unpack_wds_SLQSSetWdsEventCallback_ind_t, 592
currDBTechAvail, 593
currNWInfo, 593
dBTechAvail, 593
dBTechnology, 594
dBtechExtAvail, 594
dBtechnologyExt, 594
dataSysStatAvail, 593
dormancyStatAvail, 594
dormancyStatus, 594
mipStatus, 594
mipstatAvail, 594
netInfoLen, 594
prefNetwork, 594
ratMask, 594
rx_bytes, 594
rx_pkts, 594
soMask, 594
tx_bytes, 594
tx_pkts, 594
xferStatAvail, 594
unpack_wds_SLQSStartDataSession
wds.h, 973
unpack_wds_SLQSStartDataSession_t, 595
pFailureReason, 596
pVerboseFailReasonType, 596
pVerboseFailureReason, 596
psid, 596
unpack_wds_SLQSStopDataSession
wds.h, 974
unpack_wds_SLQSWdsGoActive
wds.h, 974
unpack_wds_SLQSWdsGoDormant
wds.h, 974
unpack_wds_SLQSWdsSetEventReport
wds.h, 974
unpack_wds_SLQSWdsSwiPDPRuntimeSettings
wds.h, 975

unpack_wds_SLQSWdsSwiPDPRuntimeSettings_t, 596
apnName, 597
bearerId, 597
contextId, 597
ipv4Address, 597
ipv4GWAddress, 597
ipv6Address, 597
ipv6GWAddress, 597
unpack_wds_SetAutoconnect
wds.h, 965
unpack_wds_SetDefaultProfile
wds.h, 965
unpack_wds_SetDefaultProfileNum
wds.h, 965
unpack_wds_SetMobileIP
wds.h, 966
unpack_wds_SetMobileIPParameters
wds.h, 966
unpack_wds_SetMobileIPProfile
wds.h, 966
unpack_wds_SetMobileIPProfile_t, 583
UnpackQmiProfileInfo
wds.h, 939
UnpackSwiAvmsEventReportBinaryUpdateSessionInfo
swiavms.h, 863
UnpackSwiAvmsEventReportConfig
swiavms.h, 864
UnpackSwiAvmsEventReportConnectionRequest
swiavms.h, 864
UnpackSwiAvmsEventReportDataSessionStatus
swiavms.h, 865
UnpackSwiAvmsEventReportHTTPStatus
swiavms.h, 865
UnpackSwiAvmsEventReportNotification
swiavms.h, 866
UnpackSwiAvmsEventReportPackageID
swiavms.h, 867
UnpackSwiAvmsEventReportRegStatus
swiavms.h, 867
UnpackSwiAvmsEventReportSessionType
swiavms.h, 867
UnpackSwiAvmsEventReportWAMSParamChange
swiavms.h, 868
unpackWdsProfileParam, 598
SlqsProfile3GPP, 598
SlqsProfile3GPP2, 598
UpdateCompleteStatus
unpack_swioma_SLQSOMADMGetSessionInfo_t, 537
updateCompleteStatus
unpack_omaDmFotaTlv_t, 496
upinRetries
slotInf, 355
uim_slotInfo, 398
upinState
slotInf, 355
uim_slotInfo, 398
upukRetries
slotInf, 355
uim_slotInfo, 398
usageMask
loc_sensorDataUsage, 97
userInputReq
unpack_omaDmConfigTlv_t, 494
unpack_omaDmFotaTlv_t, 496
userInputTimeout
unpack_omaDmConfigTlv_t, 494
unpack_omaDmFotaTlv_t, 496
Username
unpack_wds_SLQSGetRuntimeSettings_t, 591
username
unpack_wds_GetDefaultProfile_t, 577
usersize
unpack_wds_GetDefaultProfile_t, 577
ussDCS
pack_voice_SLQSOriginateUSSD_t, 311
voice_USSInfo, 633
ussData
pack_voice_SLQSOriginateUSSD_t, 310
voice_USSInfo, 633
ussLen
pack_voice_SLQSOriginateUSSD_t, 311
voice_USSInfo, 633
uusInfo
voice_allCallsUUSInfo, 600

VDOP
loc_precisionDilution, 94
val
unpack_qos_IPv6TrafCls_t, 501
unpack_qos_Tos_t, 518
val_3GPP2Pri
unpack_qos_swiQosFlow_t, 517
val_3GPPImCn
unpack_qos_swiQosFlow_t, 517
val_3GPPResResidualBER
unpack_qos_swiQosFlow_t, 517
val_3GPPSigInd
unpack_qos_swiQosFlow_t, 517
val_3GPPTraHdlPri
unpack_qos_swiQosFlow_t, 517
validMask
loc_satelliteInfo, 96
ValidityCW0
unpack_nas_SLQSSwiGetLteCQI_t, 489
ValidityCW1
unpack_nas_SLQSSwiGetLteCQI_t, 489
value
pack_dms_UIMSetPINProtection_t, 233
pack_dms_UIMVerifyPIN_t, 235
value_length
DMScustSettingInfo, 51
pack_dms_SetCustFeaturesV2_t, 227
verboseSessnEndReason
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 592

verboseSessnEndReasonType
unpack_wds_SLQSSetPacketSrvStatusCallback_t, 592
verifyLeft
uim_personalizationStatus, 390
uim_remainingRetries, 394
verifyPIN
pack_uim_VerifyPin_t, 309
verifyRetriesLeft
unpack_dms_UIMGetControlKeyStatus_t, 430
unpack_dms_UIMSetControlKeyProtection_t, 433
unpack_dms_UIMSetPINProtection_t, 433
version
unpack_omaDmFotaTlv_t, 496
unpack_qos_swiQosFilter_t, 513
versionString
pack_dms_SLQSSwiSetOSInfo_t, 230
unpack_dms_SLQSSwiGetOSInfo_t, 427
versionlength
unpack_omaDmFotaTlv_t, 496
vertConfidence
pack_loc_SLQSLOCInjectPosition_t, 246
vertReliability
pack_loc_SLQSLOCInjectPosition_t, 246
vertUnc
pack_loc_SLQSLOCInjectPosition_t, 246
VirtStream
nas_protocolSubtypeElement, 177
voice.h, 907
MAXVOICEUSSDLENGTH, 911
pack_voice_AnswerUSSD, 911
pack_voice_CancelUSSD, 911
pack_voice_OriginateUSSD, 912
pack_voice_SLQSOriginateUSSD, 912
pack_voice_SLQSVoiceALSSelectLine, 913
pack_voice_SLQSVoiceALSSetLineSwitching, 913
pack_voice_SLQSVoiceAnswerCall, 913
pack_voice_SLQSVoiceBindSubscription, 914
pack_voice_SLQSVoiceBurstDTMF, 914
pack_voice_SLQSVoiceDialCall, 914
pack_voice_SLQSVoiceEndCall, 915
pack_voice_SLQSVoiceGetAllCallInfo, 915
pack_voice_SLQSVoiceGetCLIP, 917
pack_voice_SLQSVoiceGetCLIR, 917
pack_voice_SLQSVoiceGetCNAP, 918
pack_voice_SLQSVoiceGetCOLP, 918
pack_voice_SLQSVoiceGetCOLR, 918
pack_voice_SLQSVoiceGetCallBarring, 915
pack_voice_SLQSVoiceGetCallForwardingStatus, 916
pack_voice_SLQSVoiceGetCallInfo, 916
pack_voice_SLQSVoiceGetCallWaiting, 917
pack_voice_SLQSVoiceGetConfig, 919
pack_voice_SLQSVoiceIndicationRegister, 919
pack_voice_SLQSVoiceManageCalls, 919
pack_voice_SLQSVoiceOrigUSSDNoWait, 920
pack_voice_SLQSVoiceSendFlash, 920
pack_voice_SLQSVoiceSetCallBarringPassword, 920
pack_voice_SLQSVoiceSetConfig, 921
pack_voice_SLQSVoiceSetPreferredPrivacy, 921
pack_voice_SLQSVoiceSetSUPSService, 922
pack_voice_SLQSVoiceStartContDTMF, 922
pack_voice_SLQSVoiceStopContDTMF, 922
unpack_voice_AnswerUSSD, 923
unpack_voice_CancelUSSD, 923
unpack_voice_OriginateUSSD, 923
unpack_voice_SLQSOriginateUSSD, 924
unpack_voice_SLQSVoiceALSSelectLine, 924
unpack_voice_SLQSVoiceALSSetLineSwitching, 924
unpack_voice_SLQSVoiceAnswerCall, 925
unpack_voice_SLQSVoiceBindSubscription, 925
unpack_voice_SLQSVoiceBurstDTMF, 925
unpack_voice_SLQSVoiceDialCall, 926
unpack_voice_SLQSVoiceEndCall, 926
unpack_voice_SLQSVoiceGetAllCallInfo, 926
unpack_voice_SLQSVoiceGetCLIP, 928
unpack_voice_SLQSVoiceGetCLIR, 929
unpack_voice_SLQSVoiceGetCNAP, 929
unpack_voice_SLQSVoiceGetCOLP, 929
unpack_voice_SLQSVoiceGetCOLR, 930
unpack_voice_SLQSVoiceGetCallBarring, 927
unpack_voice_SLQSVoiceGetCallForwarding-Status, 927
unpack_voice_SLQSVoiceGetCallInfo, 928
unpack_voice_SLQSVoiceGetCallWaiting, 928
unpack_voice_SLQSVoiceGetConfig, 930
unpack_voice_SLQSVoiceIndicationRegister, 930
unpack_voice_SLQSVoiceManageCalls, 931
unpack_voice_SLQSVoiceOrigUSSDNoWait, 931
unpack_voice_SLQSVoiceSendFlash, 931
unpack_voice_SLQSVoiceSetCallBarringPassword, 932
unpack_voice_SLQSVoiceSetConfig, 932
unpack_voice_SLQSVoiceSetPreferredPrivacy, 932
unpack_voice_SLQSVoiceSetSUPSService, 933
unpack_voice_SLQSVoiceStartContDTMF, 933
unpack_voice_SLQSVoiceStopContDTMF, 933
voice_CLIPResp, 616
ActiveStatus, 617
ProvisionStatus, 617
voice_CLIRResp, 617
ActiveStatus, 617
ProvisionStatus, 617
voice_CNAPResp, 617
ActiveStatus, 618
ProvisionStatus, 618
voice_COLPResp, 618
ActiveStatus, 619
ProvisionStatus, 619
voice_COLRResp, 619
ActiveStatus, 619
ProvisionStatus, 619

voice_CUGInfo, 621
CUGIndex, 622
SuppOA, 622
SuppPrefCUG, 622
voice_DTMFLengths, 623
DTMFInterdigitInterval, 624
DTMFPulseWidth, 624
voice_USSInfo, 633
ussDCS, 633
ussData, 633
ussLen, 633
voice_UUSInfo, 633
UUSData, 634
UUSDatalen, 634
UUSDcs, 634
UUSType, 634
voice_airTimer, 598
airTimerValue, 598
namID, 598
voice_allCallsAlphaIDInfo, 598
AlphaIDInfo, 599
callID, 599
voice_allCallsDiagInfo, 599
callID, 599
DiagInfo, 599
voice_allCallsUUSInfo, 599
callID, 600
uusInfo, 600
voice_alphaIDInfo, 600
alphaDcs, 601
alphaLen, 601
alphaText, 601
voice_arrAlertingPattern, 601
alertingPattern, 601
callID, 601
numInstances, 601
voice_arrAlertingType, 601
AlertingType, 602
callID, 602
numInstances, 602
voice_arrAlphaID, 602
allCallsAlphaIDInfoArr, 603
numInstances, 603
voice_arrCallEndReason, 603
callEndReason, 604
callID, 604
numInstances, 604
voice_arrCallInfo, 604
getAllCallInfo, 604
numInstances, 604
voice_arrCalledPartyNum, 603
CalledPartyNum, 603
numInstances, 603
voice_arrConnectPartyNum, 604
ConnectedPartyNum, 605
numInstances, 605
voice_arrDiagInfo, 605
DiagInfo, 605
numInstances, 605
voice_arrRedirPartyNum, 605
numInstances, 606
RedirPartyNum, 606
voice_arrRemotePartyName, 606
GetAllCallRmtPtyName, 606
numInstances, 606
voice_arrRemotePartyNum, 607
numInstances, 607
RmtPtyNum, 607
voice_arrSvcOption, 607
callID, 608
numInstances, 608
srvOption, 608
voice_arrUUSInfo, 608
AllCallsUUSInfo, 608
numInstances, 608
voice_burstDTMFInfo, 608
digitCnt, 609
pCallID, 609
pDigitBuff, 609
voice_callFWExtInfo, 611
noReplyTimer, 612
numLen, 612
numPlan, 613
numType, 613
number, 612
PI, 613
SI, 613
SvcClass, 613
SvcStatus, 613
voice_callFWInfo, 613
noReplyTimer, 613
numLen, 614
number, 613
SvcClass, 614
SvcStatus, 614
voice_callFwdTypeAndPlan, 610
numberPlan, 611
numberType, 611
voice_callInfo, 614
callID, 615
callState, 615
callType, 615
direction, 615
mode, 615
voice_calledPartySubAdd, 609
extBit, 610
oddEvenInd, 610
subAddr, 610
subAddrLen, 610
subAddrType, 610
voice_ccSUPSType, 615
reason, 616
svcType, 616
voice_connectNumInfo, 620
callerID, 621
callerIDLen, 621

numPlan, 621
numPresInd, 621
numType, 621
screeningInd, 621
voice_curAMRConfig, 622
gsmAmrStat, 622
wcdmaAmrStat, 623
voice_diagInfo, 623
diagInfoLen, 623
diagnosticInfo, 623
voice_getAllCallInformation, 624
ALS, 625
Callinfo, 625
isEmpty, 625
voice_getAllCallRmtPtyName, 625
callID, 625
RemotePartyName, 625
voice_getAllCallRmtPtyNum, 625
callID, 626
RemotePartyNum, 626
voice_getCallFWExtInfo, 626
CallFWExtInfo, 626
numInstances, 626
voice_getCallFWInfo, 626
CallFWInfo, 627
numInstances, 627
voice_peerNumberInfo, 627
callID, 628
numLen, 628
numPI, 628
numPlan, 628
numSI, 628
numType, 628
number, 628
voice_prefVoiceSO, 628
evrcCapability, 630
homeOrigVoiceSO, 630
homePageVoiceSO, 630
namID, 630
roamOrigVoiceSO, 630
voice_remotePartyName, 630
callerName, 631
codingScheme, 631
nameLen, 631
namePI, 631
voice_remotePartyNum, 631
numLen, 632
presentationInd, 632
remPartyNumber, 632
voice_roamTimer, 632
namID, 633
roamTimerValue, 633
VoiceNumber
unpack_dms_GetVoiceNumber_t, 412
voiceNumberSize
unpack_dms_GetVoiceNumber_t, 412
voiceSvc
pack_voice_SLQSVoiceSetSUPSService_t, 327
voteForInit
uim_registerRefresh, 394
wAlertMsgLength
_UnpackSwiAvmsEventReportConfig, 35
WCDMACellInfo
nas_lteWcdmaCellInfo, 157
WCDMAECIOThreshListLen
nas_WCDMAECIOThresh, 208
WCDMARSSIThreshListLen
nas_WCDMARSSIThresh, 209
WCDMASSInfo
unpack_nas_SLQSNasGetSigInfo_t, 481
wDescriptionLength
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
wErrorCode
_UnpackSwiAvmsEventReportDataSessionStatus, 36
wHTTPStatus
_UnpackSwiAvmsEventReportHTTPStatus, 37
wNameLength
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
WORD
SwiDataTypes.h, 878
wSessionStatus
_UnpackSwiAvmsEventReportNotification, 39
wUpdateCompeteStatus
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
wUserInputTimeout
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
_UnpackSwiAvmsEventReportConfig, 35
_UnpackSwiAvmsEventReportConnectionRequest, 35
wVersionLength
_UnpackSwiAvmsEventReportBinaryUpdate-SessionInfo, 34
wWamsChangeMask
_UnpackSwiAvmsEventReportWAMSParam-Change, 40
wake_mask
pack_rms_SetSMSWake_t, 279
unpack_rms_GetSMSWake_t, 519
wcdmaAmrStat
voice_curAMRConfig, 623
wcdmaRRCState
nas_WCDMAInfoLTENeighborCell, 209
wds.h, 934
BYT_STAT_STAT_MASK, 939
PACK_WDS_IPV4, 939
PACK_WDS_IPV6, 939
pack_wds_DHCPv4ClientLeaseChange, 940
pack_wds_GetAutoconnect, 940
pack_wds_GetByteTotals, 940
pack_wds_GetConnectionRate, 941
pack_wds_GetDataBearerTechnology, 941

pack_wds_GetDefaultProfile, 942
pack_wds_GetDefaultProfileNum, 942
pack_wds_GetDormancyState, 942
pack_wds_GetLastMobileIPError, 943
pack_wds_GetMobileIP, 943
pack_wds_GetMobileIPProfile, 944
pack_wds_GetPacketStatistics, 944
pack_wds_GetPacketStatus, 944
pack_wds_GetSessionDuration, 945
pack_wds_GetSessionState, 945
pack_wds_RMSetTransferStatistics, 946
pack_wds_SLQSCreateProfile, 949
pack_wds_SLQSDeleteProfile, 949
pack_wds_SLQSGet3GPPConfigItem, 949
pack_wds_SLQSGetCurrDataSystemStat, 950
pack_wds_SLQSGetCurrentChannelRate, 950
pack_wds_SLQSGetDUNCallInfo, 951
pack_wds_SLQSGetDataBearerTechnology, 951
pack_wds_SLQSGetProfileSettings, 952
pack_wds_SLQSGetRuntimeSettings, 952
pack_wds_SLQSModifyProfile, 952
pack_wds_SLQSResetPacketStatics, 953
pack_wds_SLQSSGetDHCPv4ClientConfig, 954
pack_wds_SLQSSGetLoopback, 955
pack_wds_SLQSSSetDHCPv4ClientConfig, 955
pack_wds_SLQSSSetLoopback, 956
pack_wds_SLQSSet3GPPConfigItem, 953
pack_wds_SLQSSetIPFamilyPreference, 954
pack_wds_SLQSSetWdsEventCallback, 954
pack_wds_SLQSStartDataSession, 956
pack_wds_SLQSStopDataSession, 956
pack_wds_SLQSWdsGoActive, 957
pack_wds_SLQSWdsGoDormant, 957
pack_wds_SLQSWdsSetEventReport, 958
pack_wds_SLQSWdsSwiPDPRuntimeSettings, 958
pack_wds_SetAutoconnect, 946
pack_wds_SetDefaultProfile, 946
pack_wds_SetDefaultProfileNum, 947
pack_wds_SetMobileIP, 947
pack_wds_SetMobileIPParameters, 948
pack_wds_SetMobileIPProfile, 948
unpack_wds_DHCPv4ClientLease_ind, 959
unpack_wds_DHCPv4ClientLeaseChange, 959
unpack_wds_GetAutoconnect, 959
unpack_wds_GetByteTotals, 960
unpack_wds_GetConnectionRate, 960
unpack_wds_GetDataBearerTechnology, 960
unpack_wds_GetDefaultProfile, 961
unpack_wds_GetDefaultProfileNum, 961
unpack_wds_GetDormancyState, 961
unpack_wds_GetLastMobileIPError, 962
unpack_wds_GetMobileIP, 962
unpack_wds_GetMobileIPProfile, 962
unpack_wds_GetPacketStatistics, 963
unpack_wds_GetPacketStatus, 963
unpack_wds_GetSessionDuration, 963
unpack_wds_GetSessionState, 964
unpack_wds_RMSetTransferStatistics, 964
unpack_wds_RMTransferStatistics_ind, 964
unpack_wds_SLQSCreateProfile, 967
unpack_wds_SLQSDeleteProfile, 967
unpack_wds_SLQSGet3GPPConfigItem, 967
unpack_wds_SLQSGetCurrDataSystemStat, 968
unpack_wds_SLQSGetCurrentChannelRate, 968
unpack_wds_SLQSGetDUNCallInfo, 969
unpack_wds_SLQSGetDataBearerTechnology, 968
unpack_wds_SLQSGetProfileSettings, 969
unpack_wds_SLQSGetRuntimeSettings, 969
unpack_wds_SLQSModifyProfile, 970
unpack_wds_SLQSResetPacketStatics, 970
unpack_wds_SLQSSGetDHCPv4ClientConfig, 972
unpack_wds_SLQSSGetLoopback, 972
unpack_wds_SLQSSSetDHCPv4ClientConfig, 973
unpack_wds_SLQSSSetLoopback, 973
unpack_wds_SLQSSet3GPPConfigItem, 970
unpack_wds_SLQSSetIPFamilyPreference, 970
unpack_wds_SLQSSetPacketSrvStatusCallback, 971
unpack_wds_SLQSSetWdsEventCallback, 971
unpack_wds_SLQSSetWdsEventCallback_ind, 971
unpack_wds_SLQSStartDataSession, 973
unpack_wds_SLQSStopDataSession, 974
unpack_wds_SLQSWdsGoActive, 974
unpack_wds_SLQSWdsGoDormant, 974
unpack_wds_SLQSWdsSetEventReport, 974
unpack_wds_SLQSWdsSwiPDPRuntimeSettings, 975
unpack_wds_SetAutoconnect, 965
unpack_wds_SetDefaultProfile, 965
unpack_wds_SetDefaultProfileNum, 965
unpack_wds_SetMobileIP, 966
unpack_wds_SetMobileIPParameters, 966
unpack_wds_SetMobileIPProfile, 966
UnpackQmiProfileInfo, 939
wds_DHCPLeaseOptTlv, 637
numOpt, 637
optList, 637
optListData, 637
TlvPresent, 637
wds_DHCPLeaseStateTlv, 637
leaseState, 637
TlvPresent, 637
wds_DHCPOpt, 637
optCode, 638
optValLen, 638
pOptVal, 638
wds_DHCPProfileIdTlv, 638
profileId, 638
profileType, 638
TlvPresent, 638
wds_DHCPv4HWConfig, 638
chaddr, 638

chaddrLen, 638
hwType, 639
wds_DHCPv4Option, 639
optCode, 639
optVal, 639
optValLen, 639
wds_DHCPv4OptionList, 639
numOpt, 640
pOptList, 640
wds_DHCPv4ProfileId, 640
profileId, 640
profileType, 640
wds_DataULongLongTlv, 636
TlvPresent, 636
ullData, 636
wds_DataULongTlv, 636
TlvPresent, 637
ulData, 637
wds_Domain, 640
domainLen, 641
domainName, 641
wds_DomainNameList, 641
domain, 641
numInstances, 641
wds_GPRSQoS, 641
delayClass, 642
meanThroughputClass, 642
peakThroughputClass, 642
precedenceClass, 642
reliabilityClass, 642
wds_IPV6AddressInfo, 642
IPAddressV6, 643
IPV6PrefixLen, 643
wds_IPV6GWAddressInfo, 643
gwAddressV6, 643
gwV6PrefixLen, 643
wds_IPv4AdTlv, 642
IPv4Addr, 642
TlvPresent, 642
wds_PCSCFFQDNAddress, 643
fqdnAddr, 644
fqdnLen, 644
wds_PCSCFFQDNAddressList, 644
numInstances, 644
pcsfFQDNAddress, 644
wds_PCSCFIPv4ServerAddressList, 644
numInstances, 645
pscsfIPv4Addr, 645
wds_ProfileIdentifier, 645
profileIndex, 645
profileType, 645
wds_TrStatInd, 646
statsMask, 646
statsPeriod, 646
wds_UMTSMinQoS, 646
deliveryErrSDU, 648
grntDownlinkBitrate, 648
grntUplinkBitrate, 649
maxDownlinkBitrate, 649
maxSDUSize, 649
maxUplinkBitrate, 649
qosDeliveryOrder, 649
resBerRatio, 649
sduErrorRatio, 649
trafficClass, 649
trafficPriority, 649
transferDelay, 649
wds_currNetworkInfo, 635
NetworkType, 636
RATMask, 636
SOMask, 636
wds_profileInfo, 645
SlqsProfile3GPP, 646
SlqsProfile3GPP2, 646
wdsDhcpv4HwConfig, 649
chaddr, 649
chaddrLen, 649
hwType, 649
wdsDhcpv4Option, 649
optCode, 650
optVal, 650
optValLen, 650
wdsDhcpv4OptionList, 650
numOpt, 650
pOptList, 650
wdsDhcpv4ProfileId, 650
profileId, 651
profileType, 651

xAxis
sensorData_t, 352
xferStatAvail
unpack_wds_SLQSSetWdsEventCallback_ind_t, 594
xid
pack_loc_SLQSLOCGetBestAvailPos_t, 241
pack_qmi_t, 277
unpack_qmi_t, 499

yAxis
sensorData_t, 352
year
nas_timeInfo, 200
nas_UniversalTime, 206

zAxis
sensorData_t, 352