QMI DMS 1.27

QMI Device Management Svc Spec

80-VB816-4 T

August 9, 2013

Confidential and Proprietary - Qualcomm Technologies, Inc.

NO PUBLIC DISCLOSURE PERMITTED: Please report postings of this document on public servers or websites to: DocCtrlAgent@qualcomm.com.

Restricted Distribution. Not to be distributed to anyone who is not an employee of either Qualcomm or its subsidiaries without the express approval of Qualcomm's Configuration Management.

Not to be used, copied, reproduced, or modified in whole or in part, nor its contents revealed in any manner to others without the express written permission of Qualcomm Technologies, Inc.

Qualcomm reserves the right to make changes to the product(s) or information contained herein without notice. No liability is assumed for any damages arising directly or indirectly by their use or application. The information provided in this document is provided on an "as is" basis.

This document contains confidential and proprietary information and must be shredded when discarded.

Qualcomm and MSM are trademarks of QUALCOMM Incorporated, registered in the United States and other countries. All QUALCOMM Incorporated trademarks are used with permission. Other product and brand names may be trademarks or registered trademarks of their respective owners.

This technical data may be subject to U.S. and international export, re-export, or transfer ("export") laws. Diversion contrary to U.S. and international law is strictly prohibited.

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

Contents

List of Tables

Revision History

Revision	Date	Description
A	Apr 2006	Initial release.
B	Sep 2007	Added commands for UIM PIN operations; corrected error in TLV types in Sections 3.9.2 and 3.10.2
C	May 2008	Added commands to get the hardware revision and manage device operating mode
D	Dec 2009	Updated QMI DMS revision to 1.3; added new messages for device time, band capability, UIM state, ICCID, IMSI, mobile ID, PRL version, User Lock State/Code, validating SPC, and SIM lock (Control Key) operations
E	Apr 2010	Added Persistent Low Power mode to operating mode messages; updated Sections 3.3, 3.17.2, 3.17.3, 3.18.1, and 3.18.3
F	May 2010	Updated QMI DMS version to 1.4; extended radio interface list in QMI_DMS_GET_DEVICE_CAP for LTE; new TLVS added to QMI_DMS_GET_TIME; new message added to QMI_DMS_SET_TIME
G	Feb 2011	Updated QMI DMS version to 1.5; added new message for software version information; set SPC, and added additional TLV for GET Device Serial Numbers to have the IMEI SVN; added additional TLV to PRL_VER for PRL_ONLY preference
H	Feb 2011	Minor version number corrected; no other changes have been made to content
J	Mar 2011	Updated QMI DMS version to 1.6; updated table 2.5.2, 3.1; added Activation State and Wireless Disable State TLVs in 3.2.2/3.3, Boot Code and PRI Revision TLVs in 3.7.2, IMSI TLV in 3.8.2, Offline Reason and Hardware Restricted Mode TLVs in 3.17.2; added sections 3.20, 3.21, 3.22, 3.27, 3.28, 3.29, 3.30, 3.39, 3.43, 3.44
K	Nov 2011	Updates to this version include minor version 7 and 8. • Added PRL Init TLV in QMI_DMS_SET_EVENT_REPORT and QMI_DMS_EVENT_REPORT_IND • Updated the description of QMI_DMS_GET_PRL_VER • Added LTE Band Capability TLV in QMI_DMS_GET_BAND_CAPABILITY • Added Appendix A and Appendix B
L	Feb 2012	Added QMI_DMS message QMI_DMS_GET_CURRENT_PRL_INFO.
M	Apr 2012	Updates for this version include QMI_DMS minor version 10 and minor version 11. • Added optional TLVs in QMI_DMS_GET_DEVICE_CAP • Updated mandatory TLVs in QMI_DMS_GET_DEVICE_CAP • Added TDS Band Capability TLV in QMI_DMS_GET_BAND_CAPABILITY Added the following information to TLV tables: • Version first introduced • Field type
N	May 2012	Updates for this revision include minor version 12. Updated mandatory Operating Mode TLV (Sections 3.2.3 and 3.16.2). Updated Section 3.16.3.
P	Sep 2012	Updates for this revision include minor version 13. Updated Section 2.3.1. Added new TLV Simultaneous Voice and Data Capability (Section 3.3.2).

Revision	Date	Description
R	Jan 2013	Updates for this revision include minor version 14 through minor version 18. Updated Table 1-1 Reference documents and standards. Deprecated messages: QMI_DMS_UIM_SET_PIN_PROTECTION, QMI_DMS_UIM_VERIFY_PIN, QMI_DMS_UIM_UNBLOCK_PIN, QMI_DMS_UIM_CHANGE_PIN, QMI_DMS_UIM_GET_PIN_STATUS, QMI_DMS_UIM_GET_ICCID, QMI_DMS_UIM_GET_CK_STATUS, QMI_DMS_UIM_SET_CK_PROTECTION, QMI_DMS_UIM_UNBLOCK_CK, QMI_DMS_UIM_GET_IMSI, and QMI_DMS_UIM_GET_STATE. Added Appendix C Deprecated QMI_DMS Messages. Added new messages: • QMI_DMS_GET_SUPPORTED_MSGS (Section 3.3) • QMI_DMS_GET_SUPPORTED_FIELDS (Section 3.4) • QMI_DMS_BIND_SUBSCRIPTION (Section 3.47) • QMI_DMS_GET_BIND_SUBSCRIPTION (Section 3.48)
T	Aug 2013	Updates for this revision include minor version 19 through minor version 27. Updated sections 3.19.2 and 3.5.3, and Appendix B. Updated TLVs: • Bind Subscription (Section 3.47.1) • Bound Subscription (Section 3.48.2) Added new TLVs: • CDMA Lock Mode Reporting (Section 3.2.1) • CDMA Lock Mode State (Section 3.2.3) • Device Multisim Capability (Section 3.5.2) Added new messages: • QMI_DMS_SET_AP_SW_VERSION (Section 3.49) • QMI_DMS_GET_CDMA_LOCK_MODE (Section 3.50) • QMI_DMS_SET_TEST_CONFIG (Section 3.51) • QMI_DMS_GET_TEST_CONFIG (Section 3.52) • QMI_DMS_CLEAR_TEST_CONFIG (Section 3.53)

Note: There is no Rev. I, O, Q, S, X, or Z per Mil. standards.

1 Introduction

1.1 Purpose

This specification documents Major Version 1 of the Qualcomm Messaging Interface (QMI) Wireless Data Service (QMI_DMS).

The QMI_DMS provides applications running on a tethered device, i.e., Terminal Equipment (TE), with the following commands related to device management:

- Device identification (manufacturer, model, firmware revision, phone number, serial number)
- Device capabilities (data service type, SIM, data rate)
- Device power state (battery level, power source)

It is expected that user-level applications, e.g., connection managers and/or device drivers on the TE, use QMI_DMS to access this functionality on the MSM™ device.

1.2 Scope

This document is intended for software developers who are developing code to interact with the Qualcomm MSM device from a Host processor for IP connectivity-related operations.

This document provides the following details about the QMI_DMS:

- Theory of operation – Chapter 2 provides the theory of operation of QMI_DMS. This chapter includes messaging conventions, assigned QMI service type, fundamental service concepts, and state variables related to the service.
- Message formats, syntax, and semantics – Chapter 3 provides the specific syntax and semantics of messages included in this version of the QMI_DMS specification.
- Additional information – Appendix A through Appendix C provide tables for band capability, LTE band capability, and a list of deprecated messages.

1.3 Conventions

Function declarations, function names, type declarations, and code samples appear in a different font. For example, `#include`.

An asterisk (*) in a TLV indicates that it is applicable only for 3GPP2.

A double asterisk (**) in a TLV indicates that it is applicable only for 3GPP.

Parameter types are indicated by arrows:

→	Designates an input parameter
←	Designates an output parameter
↔	Designates a parameter used for both input and output

1.4 References

Reference documents are listed in Table 1-1. Reference documents that are no longer applicable are deleted from this table; therefore, reference numbers may not be sequential.

Table 1-1 Reference documents and standards

Ref.	Document	
Qualcomm Technologies		
Q1	*Application Note: Software Glossary for Customers*	CL93-V3077-1
Q2	*Qualcomm MSM Interface (QMI) Architecture*	80-VB816-1
Q3	*Multimode GSDI SIM, USIM, and R-UIM Interface Specification and Operational Description*	80-V5329-1
Q4	*QMI UIM, QMI User Identity Module Spec*	80-VB816-12
Standards		
S1	*3rd Generation Partnership Project; Technical Specification Group Terminals; AT command set for User Equipment (UE) (Release 1999)*	3GPP TS 27.007
S2	*Data Service Options for Spread Spectrum Systems: AT Command Processing and the Rm Interface*	3GPP2 C.S0017-003-A
S3	*Data Transmission Systems and Equipment – Extensions to Serial Asynchronous Dialing and Control*	TIA/EIA/IS-131
S4	*3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Characteristics of the USIM Application*	3GPP TS 31.102
S5	*3rd Generation Partnership Project; Technical Specification Group Terminals; Specification of the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface*	3GPP TS 51.011
S6	*3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Personalization of Mobile Equipment (ME); Mobile functionality specification (Release 5)*	3GPP TS 22.022
S7	*Over-the-Air Service Provisioning of Mobile Stations in Spread Spectrum Standards*	TIA/EIA/IS-683

1.5 Technical Assistance

For assistance or clarification on information in this document, submit a case to Qualcomm Technologies at https://support.cdmatech.com.

If you do not have access to the CDMATech Support website, register for access or send email to support.cdmatech@qti.qualcomm.com.

1.6 Acronyms

For definitions of terms and abbreviations, refer to [Q1]. Table 1-2 lists terms that are specific to this document.

Table 1-2 Acronyms

Acronym	Definition
AP	application processor
CK	control key
CS	circuit-switched
DMS	Device Management Service
ERI	extended roaming indicator
ESN	electronic serial number
HDR	high data range
ICCID	integrated circuit card ID
IMEI	international mobile equipment identity
IMSI	international mobile station identity
LTE	long term evolution
MDN	mobile directory number
MEID	mobile equipment identifier
MSISDN	mobile station international subscriber directory number
Multisim	multiple simultaneous (active radio interfaces)
NAM	number assignment module
OTASP	over-the-air service programming
PRI	product release information
PRL	preferred roaming list
PS	packet-switched
PUK	PIN unlock key
QMI	Qualcomm Messaging Interface
SKU	stock keeping unit
SIM	subscriber identity module
SPC	service programming code
SVN	software version number
TDS	test data service
TLV	type-length-value
UIM	user identity module

2 Theory of Operation

2.1 Generalized QMI Service Compliance

The QMI_DMS service complies with the generalized QMI service specification, including the rules for messages, indications and responses, byte ordering, arbitration, constants, result, and error code values described in [Q2]. Extensions to the generalized QMI service theory of operation are noted in subsequent sections of this chapter.

2.2 DMS Service Type

The DMS is assigned QMI service type 0x02.

2.3 Message Definition Template

2.3.1 Response Message Result TLV

This Type-Length-Value (TLV) is present in all Response messages defined in this document. It is not present in the Indication messages.

Name	Version introduced	Version last modified
Result Code	Corresponding command's *Version introduced*	N/A

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x02			1	Result Code
Length	4			2	
Value	→	uint16	qmi_result	2	Result code • QMI_RESULT_SUCCESS • QMI_RESULT_FAILURE
		uint16	qmi_error	2	Error code – Possible error code values are described in the error codes section of each message definition

2.4 QMI_DMS Fundamental Concepts

The QMI_DMS service enables the control points to query device identification-related information. Available information includes:

- Manufacturer name, device model ID, software and hardware revision
- Voice and network identification number of the device
- Device capabilities, including the maximum channel rates, data service, SIM support, and radio technologies supported
- Device serial numbers corresponding to the wireless technologies supported by the device
- Power status information (power source, battery level, etc.)
- UIM-related functions (verify, change, unblock pins, and set pin protection)
- Device time

The QMI_DMS service also enables additional device management functionality. This includes:

- Managing the operating mode of the device
- User-controlled persistent lock state and code maintained by the device

The control point can generally obtain the above information via a polling mechanism (Request and Response messages). The power status change can also be reported via asynchronous indications. These are generated on a change in the value for all parameters, except the battery level. The notification of a battery level change is reported only when a threshold percentage (specified by a control point) is passed. These event-reporting settings registered by the control point are stored in the control point's service state variables. The Reset command can be used to clear these settings, restoring them to their default values. The details for UIM and PIN-related terms used in Sections 3.11 through 3.14 of this document are located in [S4], [S5], and [Q3]. The QMI_DMS UIM commands are only supported for 3GPP devices.

2.5 Service State Variables

2.5.1 Shared State Variables

No QMI_DMS state variables are shared across control points.

2.5.2 State Variables Per Control Point

Name	Description	Possible values	Default value
report_power_state	Indicates whether a power state change is reported to the control point	• TRUE • FALSE	FALSE
battery_lvl_lower_limit	Specifies the battery strength value (as %) below which a report to the control point is sent	0 to 100	0
battery_lvl_upper_limit	Battery strength value (as %) above which a report to the control point is sent	0 to 100	100

Name	Description	Possible values	Default value
report_activation_state	Indicates whether a change in data-bearer technology is reported to the control point	• TRUE • FALSE	FALSE
report_operating_mode	Indicates whether an operating mode change is reported to the control point	• TRUE • FALSE	FALSE
report_uim_state	Indicates whether a UIM state change is reported to the control point	• TRUE • FALSE	FALSE
report_wireless_disable	Indicates whether a wireless disable state change is reported to the control point	• TRUE • FALSE	FALSE
report_prl_init	Indicates whether a PRL initialized notification is reported to the control point	• TRUE • FALSE	FALSE

3 QMI_DMS Messages

Table 3-1 QMI_DMS messages

Command	ID	Description
QMI_DMS_RESET	0x0000	Resets the DMS state variables of the requesting control point.
QMI_DMS_SET_EVENT_REPORT	0x0001	Sets the device management state reporting conditions for the requesting control point.
QMI_DMS_GET_SUPPORTED_MSGS	0x001E	Queries the set of messages implemented by the currently running software.
QMI_DMS_GET_SUPPORTED_FIELDS	0x001F	Queries the fields supported for a single command as implemented by the currently running software.
QMI_DMS_GET_DEVICE_CAP	0x0020	Requests the device capabilities.
QMI_DMS_GET_DEVICE_MFR	0x0021	Requests the device the manufacturer information.
QMI_DMS_GET_DEVICE_MODEL_ID	0x0022	Requests the device model identification.
QMI_DMS_GET_DEVICE_REV_ID	0x0023	Requests the device firmware revision identification.
QMI_DMS_GET_MSISDN	0x0024	Requests the assigned voice number.
QMI_DMS_GET_DEVICE_SERIAL_NUMBERS	0x0025	Requests the serial numbers of the device.
QMI_DMS_GET_POWER_STATE	0x0026	Requests the power status of the device.
QMI_DMS_UIM_SET_PIN_PROTECTION	0x0027	Enables or disables protection of UIM contents by a specified PIN. (Deprecated)
QMI_DMS_UIM_VERIFY_PIN	0x0028	Verifies the PIN before accessing the UIM contents. (Deprecated)
QMI_DMS_UIM_UNBLOCK_PIN	0x0029	Unblocks a blocked PIN. (Deprecated)
QMI_DMS_UIM_CHANGE_PIN	0x002A	Changes the PIN value. (Deprecated)
QMI_DMS_UIM_GET_PIN_STATUS	0x002B	Gets the status of a PIN. (Deprecated)
QMI_DMS_GET_DEVICE_HARDWARE_REV	0x002C	Queries the hardware revision of the device.

Table 3-1 QMI_DMS messages (cont.)

Command	ID	Description
QMI_DMS_GET_OPERATING_MODE	0x002D	Queries the current operating mode of the device.
QMI_DMS_SET_OPERATING_MODE	0x002E	Sets the operating mode of the device.
QMI_DMS_GET_TIME	0x002F	Queries the current time of the device.
QMI_DMS_GET_PRL_VER	0x0030	Queries the version of the active Preferred Roaming List (PRL) of the device.
QMI_DMS_GET_ACTIVATION_STATE	0x0031	Queries the activation state of the device.
QMI_DMS_ACTIVATE_AUTOMATIC	0x0032	Requests that the device perform automatic service activation.
QMI_DMS_ACTIVATE_MANUAL	0x0033	Requests that the device perform manual service activation.
QMI_DMS_GET_USER_LOCK_STATE	0x0034	Queries the state of the user lock maintained by the device.
QMI_DMS_SET_USER_LOCK_STATE	0x0035	Sets the user lock state maintained by the device.
QMI_DMS_SET_USER_LOCK_CODE	0x0036	Sets the user lock code maintained by the device.
QMI_DMS_READ_USER_DATA	0x0037	Queries the user data maintained by the device.
QMI_DMS_WRITE_USER_DATA	0x0038	Writes user data maintained by the device.
QMI_DMS_READ_ERI_FILE	0x0039	Queries the Extended Roaming Indicator (ERI) file stored on the device.
QMI_DMS_RESTORE_FACTORY_DEFAULTS	0x003A	Requests that the device reset all settings to factory defined values.
QMI_DMS_VALIDATE_SERVICE_ PROGRAMMING_CODE	0x003B	Requests the device to validate a specified service programming code.
QMI_DMS_UIM_GET_ICCID	0x003C	Queries the Integrated Circuit Card ID (ICCID) of the UIM for the device. (Deprecated)
QMI_DMS_UIM_GET_CK_STATUS	0x0040	Queries the status of a UIM facility control key. (Deprecated)
QMI_DMS_UIM_SET_CK_PROTECTION	0x0041	Sets the protection of a UIM facility control key. (Deprecated)
QMI_DMS_UIM_UNBLOCK_CK	0x0042	Unblocks a UIM facility control key. (Deprecated)
QMI_DMS_UIM_GET_IMSI	0x0043	Queries the International Mobile Station Identity (IMSI) of the UIM for the device. (Deprecated)
QMI_DMS_UIM_GET_STATE	0x0044	Queries the state of the UIM. (Deprecated)

Table 3-1 QMI_DMS messages (cont.)

Command	ID	Description
QMI_DMS_GET_BAND_CAPABILITY	0x0045	Queries the band capability of the device.
QMI_DMS_GET_FACTORY_SKU	0x0046	Queries the factory provisioned Stock Keeping Unit (SKU).
QMI_DMS_SET_TIME	0x004B	Sets the time on the device.
QMI_DMS_GET_ALT_NET_CONFIG	0x004D	Queries the alternative network interface configuration used for the device.
QMI_DMS_SET_ALT_NET_CONFIG	0x004E	Sets the alternative network interface configuration used for the device.
QMI_DMS_GET_SW_VERSION	0x0051	Queries the software version from the device.
QMI_DMS_SET_SPC	0x0052	Changes the service programming code of the device after authentication.
QMI_DMS_GET_CURRENT_PRL_INFO	0x0053	Queries the currently active PRL information of the device.
QMI_DMS_BIND_SUBSCRIPTION	0x0054	Associates the requesting control point with the requested subscription.
QMI_DMS_GET_BIND_SUBSCRIPTION	0x0055	Queries the subscription associated with the control point.
QMI_DMS_SET_AP_SW_VERSION	0x0056	Sets the AP software version on the modem required for an Auto Register Short message.
QMI_DMS_GET_CDMA_LOCK_MODE	0x0057	Requests the CDMA Lock mode status.
QMI_DMS_SET_TEST_CONFIG	0x0058	Sets the configuration type used while testing.
QMI_DMS_GET_TEST_CONFIG	0x0059	Gets the configuration type used for testing.
QMI_DMS_CLEAR_TEST_CONFIG	0x005A	Resets the modem configuration to production values.

3.1 QMI_DMS_RESET

Resets the DMS state variables of the requesting control point.

DMS message ID

0x0000

Version introduced

Major - 1, Minor - 0

3.1.1 Request - QMI_DMS_RESET_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.1.2 Response - QMI_DMS_RESET_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing

3.1.3 Description of QMI_DMS_RESET REQ/RESP

This command resets the issuing control point state (see Section 2.5.2) kept by the service. As a result, each shared state variable can change depending on its arbitration policy (see Section 2.5.1). Although it is performed as one operation, this is equivalent to closing the service and reopening it again, therefore the client ID of the requesting control point does not change. The control point state variables change to their default values before the response is issued.

3.2 QMI_DMS_SET_EVENT_REPORT

Sets the device management state reporting conditions for the requesting control point.

DMS message ID

0x0001

Version introduced

Major - 1, Minor - 0

3.2.1 Request - QMI_DMS_SET_EVENT_REPORT_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
Power State Reporting	1.0	1.0
Battery Level Report Limits	1.0	1.0
PIN State Reporting	Unknown	1.1
Activation State Reporting	Unknown	1.6
Operating Mode Reporting	Unknown	1.3
UIM State Reporting	Unknown	1.3
Wireless Disable State Reporting	Unknown	1.6
PRL Init Reporting	Unknown	1.7
CDMA Lock Mode Reporting	1.24	1.24

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Power State Reporting
Length	1			2	
Value	→	boolean	report_power_state	1	Values: • 0 – Do not report • 1 – Report on change in power state
Type	0x11			1	Battery Level Report Limits
Length	2			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint8	battery_lvl_lower_limit	1	The battery level is reported to the control point if the battery level falls below this lower limit (specified as percentage of remaining battery power from 0 to 100).
		uint8	battery_lvl_upper_limit	1	The battery level is reported to the control point if the battery level rises above the upper limit (specified as percentage of remaining battery power from 0 to 100).
Type	0x12			1	PIN State Reporting
Length	1			2	
Value	→	boolean	report_pin_state	1	Values: • 0 – Do not report • 1 – Report on change in PIN state
Type	0x13			1	Activation State Reporting
Length	1			2	
Value	→	boolean	report_activation_state	1	Values: • 0 – Do not report • 1 – Report activation state changes
Type	0x14			1	Operating Mode Reporting
Length	1			2	
Value	→	boolean	report_oprt_mode_state	1	Values: • 0 – Do not report • 1 – Report operating mode changes
Type	0x15			1	UIM State Reporting
Length	1			2	
Value	→	boolean	report_uim_state	1	Values: • 0 – Do not report • 1 – Report UIM state changes
Type	0x16			1	Wireless Disable State Reporting
Length	1			2	
Value	→	boolean	report_wireless_disable_state	1	Values: • 0 – Do not report • 1 – Report wireless disable state changes
Type	0x17			1	PRL Init Reporting
Length	1			2	
Value	→	boolean	report_prl_init	1	Values: • 0 – Do not report • 1 – Report PRL initialized notification
Type	0x18			1	CDMA Lock Mode Reporting
Length	1			2	
Value	→	boolean	report_cdma_lock_mode	1	Values: • 0 – Do not report (default value) • 1 – Report CDMA Lock mode state changes

3.2.2 Response - QMI_DMS_SET_EVENT_REPORT_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

3.2.3 Indication - QMI_DMS_EVENT_REPORT_IND

Message type

Indication

Sender

Service

Indication scope

Per control point (unicast)

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
Power State	1.0	1.0
PIN 1 Status	Unknown	1.1
PIN 2 Status	Unknown	1.1
Activation State	Unknown	1.6
Operating Mode	1.3	1.12
UIM State	Unknown	1.3
Wireless Disable State	Unknown	1.6
PRL Init Notification	Unknown	1.7
CDMA Lock Mode State	1.24	1.24

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Power State
Length	2			2	
Value	→	mask8	power_status	1	Power status flags. Values: Bit 0 – Power source • 0 – Powered by battery • 1 – Powered by external source Bit 1 – Battery connected • 0 – Not connected • 1 – Connected Bit 2 – Battery charging • 0 – Not charging • 1 – Charging Bit 3 – Power fault • 0 – No power fault • 1 – Recognized power fault, calls inhibited
		uint8	battery_lvl	1	Level of the battery. Values: • 0x00 – Battery is exhausted or the mobile device does not have a battery connected • 1 through 100 (0x64) – Percentage of battery capacity remaining
Type	0x11			1	PIN 1 Status
Length	3			2	
Value	→	enum8	status	1	Current status of the PIN. Values: • 0 – PIN is not initialized • 1 – PIN is enabled, not verified • 2 – PIN is enabled, verified • 3 – PIN is disabled • 4 – PIN is blocked • 5 – PIN is permanently blocked • 6 – PIN is unblocked • 7 – PIN is changed
		uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.
Type	0x12			1	PIN 2 Status
Length	3			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	status	1	Current status of the PIN. Values: • 0 – PIN is not initialized • 1 – PIN is enabled, not verified • 2 – PIN is enabled, verified • 3 – PIN is disabled • 4 – PIN is blocked • 5 – PIN is permanently blocked • 6 – PIN is unblocked • 7 – PIN is changed
		uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.
Type	0x13			1	Activation State
Length	2			2	
Value	→	enum16	activation_state	2	Service activation state. Values: • 0x00 – Service is not activated • 0x01 – Service is activated • 0x02 – Activation connecting – Network connection is in progress for automatic activation of service • 0x03 – Activation connected – Network connection is connected for automatic activation of service • 0x04 – OTASP security is authenticated • 0x05 - OTASP NAM is downloaded • 0x06 - OTASP MDN is downloaded • 0x07 - OTASP IMSI downloaded • 0x08 - OTASP PRL is downloaded • 0x09 - OTASP SPC is downloaded • 0x0A - OTASP settings are committed
Type	0x14			1	Operating Mode
Length	1			2	
Value	→	enum8	operating_mode	1	Current operating mode. Values: • 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 – Conducting network test for GSM/WCDMA
Type	0x15			1	UIM State
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	uim_state	1	UIM state. Values: • 0x00 – UIM initialization completed • 0x01 – UIM failed • 0x02 – UIM is not present • 0xFF – UIM state is currently unavailable
Type	0x16			1	Wireless Disable State
Length	1			2	
Value	→	enum8	wireless_disable_state	1	Wireless disable state. Values: • 0x00 – Wireless disable switch is turned off • 0x01 – Wireless disable switch is turned on
Type	0x17			1	PRL Init Notification
Length	1			2	
Value	→	enum8	prl_init	1	PRL initialized. Values: • 0x01 – PRL is completely loaded into the device (could be the default PRL).
Type	0x18			1	CDMA Lock Mode State
Length	4			2	
Value	→	enum	cdma_lock_mode_state	4	CDMA Lock mode state. Values: • DMS_CDMA_LOCK_MODE_OFF (0) – Phone is not CDMA locked • DMS_CDMA_LOCK_MODE_ON (1) – Phone is CDMA locked

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	The message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Some of the TLVs, e.g., report_uim_state, are not supported because the device lacks underlying support. This error is returned even if the message contains a mix of supported and unsupported TLVs. The control point is expected to register separately for each event in such a situation.

3.2.4 Description of QMI_DMS_SET_EVENT_REPORT

The control point state variables controlling event reporting are modified according to the TLVs present in the request. The service maintains a separate set of state variables for each control point. See Section 2.5.2 for more details regarding control point state variables. Specific device management state changes are communicated to the registered DMS control point via the QMI_DMS_EVENT_REPORT_IND indicator message. The AT command equivalents to this command are AT+CMER, AT+CIND, and AT+CIEV (see [S1]).

This command is sent to specific control points when the device state corresponding to one of the previous TLVs has changed. The specific control points are those that previously registered for the corresponding state to be reported using the QMI_DMS_SET_EVENT_REPORT_REQ message.

The Power State TLV is included when any of the following occurs:

- The control point sets the battery level limits, and the battery level triggers either the upper or lower limit, i.e., the Request message.
- The control point enables power state reporting and the power state changes.
- The PIN1 Status or PIN2 Status TLVs are included when the control point has enabled PIN status reporting and the PIN status has changed.

The AT command equivalents to this command are AT+CMER, AT+CIND, and AT+CIEV (see[S1]).

- The Activation State TLV is included when the activation state of the device has changed.
- The Operating Mode TLV is included when the control point has enabled Operating Mode reporting and the operating mode of the device has changed.
- The UIM State TLV is included when the control point has enabled UIM state reporting and the UIM state of the device has changed.
- The Wireless Disable TLV is included when the wireless disable signal state for the device has changed.
- The PRL Init Notification TLV is included when the PRL is loaded to the device.

3.3 QMI_DMS_GET_SUPPORTED_MSGS

Queries the set of messages implemented by the currently running software.

DMS message ID

0x001E

Version introduced

Major - 1, Minor - 16

3.3.1 Request - QMI_DMS_GET_SUPPORTED_MSGS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.3.2 Response - QMI_DMS_GET_SUPPORTED_MSGS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Common version introduced	Common version last modified
List of Supported Messages	1.6	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	List of Supported Messages
Length	Var			2	
Value	→	uint16	supported_msgs_len	2	Number of sets of the following elements: • supported_msgs
		uint8	supported_msgs	Var	This array of uint8 is a bitmask where each bit represents a message ID, i.e., starting with the LSB, bit 0 represents message ID 0, bit 1 represents message ID 1, etc. The bit is set to 1 if the message is supported; otherwise, it is set to zero. For example, if a service supports exactly four messages with IDs 0, 1, 30, and 31 (decimal), the array (in hexadecimal) is 4 bytes [03 00 00 c0].

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_INFO_UNAVAILABLE	Information is not available

3.3.3 Description of QMI_DMS_GET_SUPPORTED_MSGS REQ/RESP

This command queries the set of messages implemented by the currently running software. This may be a subset of the messages defined in this revision of the service.

3.4 QMI_DMS_GET_SUPPORTED_FIELDS

Queries the fields supported for a single command as implemented by the currently running software.

DMS message ID

0x001F

Version introduced

Major - 1, Minor - 16

3.4.1 Request - QMI_DMS_GET_SUPPORTED_FIELDS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Common version introduced	Common version last modified
Service Message ID	1.6	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Service Message ID
Length	2			2	
Value	→	uint16	msg_id	2	ID of the command for which the supported fields are requested.

Optional TLVs

None

3.4.2 Response - QMI_DMS_GET_SUPPORTED_FIELDS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Common version introduced	Common version last modified
List of Supported Request Fields	1.6	1.6
List of Supported Response Fields	1.6	1.6
List of Supported Indication Fields	1.6	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	List of Supported Request Fields
Length	Var			2	
Value	→	uint8	request_fields_len	1	Number of sets of the following elements: • request_fields
		uint8	request_fields	Var	This field describes which optional field IDs are supported in the QMI request. The array of uint8 is a bitmask where each bit represents a field (TLV) ID. Because fields 0 to 15 (decimal) are mandatory by definition, the first bit represents field ID 16. Starting with the LSB, bit 0 represents field ID 16, bit 1 represents field ID 17, etc. The bit is set to 1 if the field ID is supported; otherwise, it is set to zero. For example, if a service supports exactly four fields with IDs 16, 17, 30, and 31 (decimal), the array (in hexadecimal) is 2 bytes [03 c0].
Type	0x11			1	List of Supported Response Fields
Length	Var			2	
Value	→	uint8	response_fields_len	1	Number of sets of the following elements: • response_fields
		uint8	response_fields	Var	This field describes which optional field IDs are supported in the QMI response. Its format is the same as request_fields.
Type	0x12			1	List of Supported Indication Fields
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint8	indication_fields_len	1	Number of sets of the following elements: • indication_fields
		uint8	indication_fields	Var	This field describes which optional field IDs are supported in the QMI indication. Its format is the same as request_fields.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_REQUESTED_NUM_ UNSUPPORTED	Requested message ID is not supported by the currently running software
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_INFO_UNAVAILABLE	Information is not available

3.4.3 Description of QMI_DMS_GET_SUPPORTED_FIELDS REQ/RESP

This command queries the fields supported for a single command as implemented by the currently running software.

If the request, response, or indication is supported for the given message ID, the corresponding optional array is included in QMI_DMS_GET_SUPPORTED_FIELDS_RESP, even if the message does not contain any optional fields. This enables the client to distinguish this case from one where the service does not support the request, response, or indication.

Examples are:

- If the specified message ID is not supported by the service, the response has qmi_result = QMI_RESULT_FAILURE and qmi_error = QMI_ERR_REQUESTED_NUM_UNSUPPORTED.
- If the specified message ID is an empty message, the response has qmi_result = QMI_RESULT_SUCCESS and qmi_error = QMI_ERR_NONE. None of the optional arrays are included.
- If the specified message ID supports the request with 0 optional fields, the response with 3 optional fields (16, 17, and 18 decimal), and does not support an indication, the response has the following:
  - qmi_result = QMI_RESULT_SUCCESS
  - qmi_error = QMI_ERR_NONE
  - request_fields array is included with length zero
  - response_fields array is included with length 1 value [07]
  - indication_fields array is not included

Trailing zero bytes are omitted from the response. For example, if the message defines 20 different fields but the response only contains 16 bits, the client is to assume the last four fields are not supported.

3.5 QMI_DMS_GET_DEVICE_CAP

Requests the device capabilities.

DMS message ID

0x0020

Version introduced

Major - 1, Minor - 0

3.5.1 Request - QMI_DMS_GET_DEVICE_CAP_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.5.2 Response - QMI_DMS_GET_DEVICE_CAP_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response. This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
Device Capabilities	Unknown	1.10

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Device Capabilities
Length	Var			2	
Value	→	uint32	max_tx_channel_rate	4	Maximum Tx transmission rate in bits per second (bps) supported by the device. The value 0xFFFFFFFF implies a rate greater than or equal to 0xFFFFFFFF (4 Gbps). In multitechnology devices, this value is the greatest rate among all supported technologies.
		uint32	max_rx_channel_rate	4	Maximum Rx transmission rate in bits per second (bps) supported by the device. The value 0xFFFFFFFF implies rate greater than or equal to 0xFFFFFFFF (4 Gbps). In multitechnology devices, this value is the greatest rate among all supported technologies.
		enum8	data_service_capability	1	Values: • 0 – No data services supported • 1 – Only circuit-switched (CS) services are supported • 2 – Only packet-switched (PS) services are supported • 3 – Simultaneous CS and PS • 4 – Nonsimultaneous CS and PS
		enum8	sim_capability	1	Values: • 1 – SIM is not supported • 2 – SIM is supported
		uint8	radio_if_list_len	1	Number of sets of the following elements: • radio_if_list
		enum8	radio_if_list	Var	List of N one-byte elements describing the radio interfaces supported by the device. Values: • 1 – CDMA2000 1X • 2 – CDMA2000 HRPD (1xEV-DO) • 4 – GSM • 5 – UMTS • 8 – LTE • 9 – TDS

Optional TLVs

Name	Version introduced	Version last modified
Device Service Capability	1.11	1.11
Voice Support Capability	1.11	1.11
Simultaneous Voice and Data Capability	1.13	1.13
Device Multisim Capability	1.22	1.22

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Device Service Capability
Length	4			2	
Value	→	enum	device_service_capability	4	Values: • 1 – Only data services are supported • 2 – Only voice services are supported • 3 – Simultaneous voice and data • 4 – Nonsimultaneous voice and data
Type	0x11			1	Voice Support Capability
Length	8			2	
Value	→	mask	voice_support_capability	8	Bitmask of voice support available on device. Values: Bit 0 – GW CSFB • 0 – Not capable • 1 – Capable Bit 1 – 1x CSFB • 0 – Not capable • 1 – Capable Bit 2 – VoLTE • 0 – Not capable • 1 – Capable
Type	0x12			1	Simultaneous Voice and Data Capability
Length	8			2	
Value	→	mask	simul_voice_and_data_capability	8	Bitmask of simultaneous voice and data support available on device. Values: • Bit 0 – SVLTE capability • Bit 1 – SVDO capability **Note:** Zero bits set means that neither of the defined capabilities are supported.
Type	0x13			1	Device Multisim Capability Device capability for supporting multiple simultaneously active radio interfaces.
Length	Var			2	
Value	→	uint8	max_subscriptions	1	The maximum number of subscriptions that can be supported simultaneously.
		uint8	subscription_config_list_len	1	Number of sets of the following elements: • max_active • subscription_list
		uint8	max_active	1	The maximum number of subscriptions listed in this configuration that can be simultaneously active. If this number is less than max_subscriptions it implies that any combination of the subscriptions in this configuration can be active and the remaining can be in standby

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	subscription_list_len	1	Number of sets of the following elements: • subscription_list
		mask	subscription_list	Var	An array of max_subscriptions entries where each entry is a mask of capabilities. The client ignores any bits in the mask that it does not recognize. Values: • DMS_SUBS_CAPABILITY_AMPS (0x00000001) – • DMS_SUBS_CAPABILITY_CDMA (0x00000002) – • DMS_SUBS_CAPABILITY_HDR (0x00000004) – • DMS_SUBS_CAPABILITY_GSM (0x00000008) – • DMS_SUBS_CAPABILITY_WCDMA (0x00000010) – • DMS_SUBS_CAPABILITY_LTE (0x00000020) – • DMS_SUBS_CAPABILITY_TDS (0x00000040) –

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.5.3 Description of QMI_DMS_GET_DEVICE_CAP REQ/RESP

This command obtains the high-level capabilities of the device. The AT command equivalent to this command is AT+GCAP (see [S1], [S2], and [S3]).

The Device Multisim Capability TLV includes a list of supported multisim configurations. Each entry in this list includes a max_active field and a max_subscriptions field. The order of the bitmask fields in this list does not correspond to any type of subscription index. Each bitmask field represents capabilities of a single subscription.

For example, consider a device where max_subscriptions is 3. One entry in the subscription configuration list has max_active = 2, with the following values :
subscription list[0] = DMS_SUBS_CAPABILITY_GSM | DMS_SUBS_CAPABILITY_WCDMA
subscription list[1] = DMS_SUBS_CAPABILITY_GSM
subscription list[2] = DMS_SUBS_CAPABILITY_GSM

This means the device supports a configuration with three GSM subscriptions, a configuration with two GSM subscriptions, and one WCDMA subscription. For any of these configurations the device supports any two being active simultaneously with any remaining subscriptions on standby.

3.6 QMI_DMS_GET_DEVICE_MFR

Requests the device the manufacturer information.

DMS message ID

0x0021

Version introduced

Major - 1, Minor - 0

3.6.1 Request - QMI_DMS_GET_DEVICE_MFR_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.6.2 Response - QMI_DMS_GET_DEVICE_MFR_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response. This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
Device Manufacturer	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Device Manufacturer
Length	Var			2	
Value	→	string	device_manufacturer	Var	String identifying the device manufacturer.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate the response

3.6.3 Description of QMI_DMS_GET_DEVICE_MFR REQ/RESP

This command returns a string identifying the device manufacturer. The AT command equivalent to this command is AT+GMI (see [S1], [S2], and [S3]).

3.7 QMI_DMS_GET_DEVICE_MODEL_ID

Requests the device model identification.

DMS message ID

0x0022

Version introduced

Major - 1, Minor - 0

3.7.1 Request - QMI_DMS_GET_DEVICE_MODEL_ID_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.7.2 Response - QMI_DMS_GET_DEVICE_MODEL_ID_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response. This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
Device Model	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Device Model
Length	Var			2	
Value	→	string	device_model_id	Var	String identifying the device model.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.7.3 Description of QMI_DMS_GET_DEVICE_MODEL_ID REQ/RESP

This command returns a string identifying the model of the device. This usually corresponds to the manufacturer's model name under which the device is marketed. The AT command equivalent to this command is AT+GMM (see [S1], [S2], and [S3]).

3.8 QMI_DMS_GET_DEVICE_REV_ID

Requests the device firmware revision identification.

DMS message ID

0x0023

Version introduced

Major - 1, Minor - 0

3.8.1 Request - QMI_DMS_GET_DEVICE_REV_ID_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.8.2 Response - QMI_DMS_GET_DEVICE_REV_ID_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response. This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
Revision ID	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Revision ID
Length	Var			2	
Value	→	string	device_rev_id	Var	String containing the device revision ID.

Optional TLVs

Name	Version introduced	Version last modified
Boot Code Revision	Unknown	1.6
PRI Revision	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Boot Code Revision
Length	Var			2	
Value	→	string	boot_code_rev	Var	String containing the boot code revision.
Type	0x11			1	PRI Revision
Length	Var			2	
Value	→	string	pri_rev	Var	String containing the device PRI revision.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.8.3 Description of QMI_DMS_GET_DEVICE_REV_ID REQ/RESP

This command returns a string identifying the firmware revision of the device. This usually corresponds to the manufacturer's software revision loaded on the device. The AT command equivalent to this command is AT+GMR (see [S1], [S2], and [S3]).

If supported by the device, one or more optional TLVs are also returned:

- Boot Code revision – Revision of the boot software used to power up the device
- PRI Revision – Revision of the factory configuration loaded to the device

3.9 QMI_DMS_GET_MSISDN

Requests the assigned voice number.

DMS message ID

0x0024

Version introduced

Major - 1, Minor - 0

3.9.1 Request - QMI_DMS_GET_MSISDN_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.9.2 Response - QMI_DMS_GET_MSISDN_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response. This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
Voice Number	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Voice Number
Length	Var			2	
Value	→	string	voice_number	Var	String containing the voice number in use by the device.

Optional TLVs

Name	Version introduced	Version last modified
Mobile ID	Unknown	1.3
International Mobile Subscriber ID	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Mobile ID
Length	Var			2	
Value	→	string	mobile_id_number	Var	String containing the mobile ID number of the device.
Type	0x11			1	International Mobile Subscriber ID
Length	Var			2	
Value	→	string	imsi	Var	String containing the international mobile subscriber ID of the device.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NOT_PROVISIONED	Device does not support voice service or the value is not provisioned in the device

3.9.3 Description of QMI_DMS_GET_MSISDN REQ/RESP

The voice number is the MDN or MSISDN assigned to the mobile. If it is available in the device provisioning, an optional mobile ID and IMSI is returned. The AT command equivalent to this command is AT+CNUM (see [S3]).

3.10 QMI_DMS_GET_DEVICE_SERIAL_NUMBERS

Requests the serial numbers of the device.

DMS message ID

0x0025

Version introduced

Major - 1, Minor - 0

3.10.1 Request - QMI_DMS_GET_DEVICE_SERIAL_NUMBERS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.10.2 Response - QMI_DMS_GET_DEVICE_SERIAL_NUMBERS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Version introduced	Version last modified
ESN	1.0	1.0
IMEI	1.0	1.0
MEID	1.0	1.0
IMEI SVN	Unknown	1.5

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	ESN
Length	Var			2	
Value	→	string	esn	Var	String containing the Electronic Serial Number (ESN) of the device.
Type	0x11			1	IMEI
Length	Var			2	
Value	→	string	imei	Var	String containing the International Mobile Equipment Identity (IMEI) of the device.
Type	0x12			1	MEID
Length	Var			2	
Value	→	string	meid	Var	String containing the Mobile Equipment Identifier (MEID) of the device.
Type	0x13			1	IMEI SVN
Length	Var			2	
Value	→	string	imeisv_svn	Var	IMEI software version number

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NOT_PROVISIONED	Device does not support voice service or the value is not provisioned in the device

3.10.3 Description of QMI_DMS_GET_DEVICE_SERIAL_NUMBERS REQ/RESP

This command returns all serial numbers assigned to the device as follows:

- ESN is included for 3GPP2 devices
- IMEI is included for 3GPP devices
- MEID is included for devices that support it, e.g., 3GPP or 3GPP2
- IMEI software version number is included for 3GPP devices

The AT command equivalent to this command is AT+GSN (see [S1], [S2], and [S3]).

3.11 QMI_DMS_GET_POWER_STATE

Requests the power status of the device.

DMS message ID

0x0026

Version introduced

Major - 1, Minor - 0

3.11.1 Request - QMI_DMS_GET_POWER_STATE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.11.2 Response - QMI_DMS_GET_POWER_STATE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response. This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
Power State	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Power State
Length	2			2	
Value	→	mask8	power_status	1	Power status flags. Values: Bit 0 – Power source • 0 – Powered by battery • 1 – Powered by external source Bit 1 – Battery connected • 0 – Not connected • 1 – Connected Bit 2 – Battery charging • 0 – Not charging • 1 – Charging Bit 3 – Power fault • 0 – No power fault • 1 – Recognized power fault, calls inhibited
		uint8	battery_lvl	1	Level of the battery. Values: • 0x00 – Battery is exhausted or the mobile device does not have a battery connected • 1 through 100 (0x64) – Percentage of battery capacity remaining

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.11.3 Description of QMI_DMS_GET_POWER_STATE REQ/RESP

This command obtains information regarding the power status of the device. The information returned is described in the Mandatory TLVs section. The external power source can be one of the following:

- Wall-mounted power source
- USB charger

The AT command equivalent to this command is AT+CBC (see [S3]).

3.12 QMI_DMS_UIM_SET_PIN_PROTECTION

Enables or disables protection of UIM contents by a specified PIN. (Deprecated)

DMS message ID

0x0027

Version introduced

Major - 1, Minor - 1

3.12.1 Request - QMI_DMS_UIM_SET_PIN_PROTECTION_REQ

Message type

Request

Sender

Control Point

Mandatory TLVs

Name	Version introduced	Version last modified
PIN Protection Information	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	PIN Protection Information
Length	Var			2	
Value	→	enum8	pin_id	1	Specifies the ID of the PIN to be enabled or disabled. Values: • 1 – PIN1 (also called PIN) • 2 – PIN2
		boolean	protection_setting_enabled	1	Specifies whether the PIN is enabled. Values: • 0 – Disable PIN • 1 – Enable PIN
		uint8	pin_value_len	1	Number of sets of the following elements: • pin_value
		uint8	pin_value	Var	Specifies the PIN value of the PIN to be enabled/disabled. The protection setting is only changed if this value is successfully verified by the SIM.

Optional TLVs

None

3.12.2 Response - QMI_DMS_UIM_SET_PIN_PROTECTION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Returned if the verify operation fails.

Name	Version introduced	Version last modified
Pin Retries Status	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Pin Retries Status
Length	2			2	
Value	→	uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NO_EFFECT	Operation had no effect
QMI_ERR_ARG_TOO_LONG	Device cannot handle the length of the PIN specified
QMI_ERR_INCORRECT_PIN	PIN specified in the request is incorrect
QMI_ERR_PIN_BLOCKED	PIN is blocked; an unblock operation needs to be issued
QMI_ERR_PIN_PERM_BLOCKED	PIN is permanently blocked; the UIM is unusable

QMI_ERR_UIM_NOT_INITIALIZED	PIN is not yet initialized because the SIM initialization has not finished; try the PIN operation later
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_INVALID_PINID	PIN specified in the request is invalid
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed

3.12.3 Description of QMI_DMS_UIM_SET_PIN_PROTECTION REQ/RESP

This command enables or disables the protection of the UIM contents by a specified PIN.

This command is deprecated. QMI_UIM_SET_PIN_PROTECTION is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.13 QMI_DMS_UIM_VERIFY_PIN

Verifies the PIN before accessing the UIM contents. (Deprecated)

DMS message ID

0x0028

Version introduced

Major - 1, Minor - 1

3.13.1 Request - QMI_DMS_UIM_VERIFY_PIN_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
PIN Value	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	PIN Value
Length	Var			2	
Value	→	enum8	pin_id	1	Specifies the ID of the PIN to be enabled or disabled. Values: • 1 – PIN1 (also called PIN) • 2 – PIN2
		uint8	pin_value_len	1	Number of sets of the following elements: • pin_value
		uint8	pin_value	Var	Specifies the PIN value of the PIN to be verified; the protection setting is only changed if this value is successfully verified by the SIM.

Optional TLVs

None

3.13.2 Response - QMI_DMS_UIM_VERIFY_PIN_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Returned if the verify operation fails.

Name	Version introduced	Version last modified
PIN Retries Status	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	PIN Retries Status
Length	2			2	
Value	→	uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NO_EFFECT	Operation had no effect
QMI_ERR_ARG_TOO_LONG	Device cannot handle the length of the PIN specified in the request
QMI_ERR_INCORRECT_PIN	PIN specified in the request is incorrect
QMI_ERR_PIN_BLOCKED	PIN is blocked; an unblock operation needs to be issued
QMI_ERR_PIN_PERM_BLOCKED	PIN is permanently blocked; the SIM is unusable

QMI_ERR_UIM_NOT_INITIALIZED	PIN is not yet initialized because the SIM initialization has not finished; try the PIN operation later
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_INVALID_PINID	PIN specified in the request is invalid
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed

3.13.3 Description of QMI_DMS_UIM_VERIFY_PIN REQ/RESP

This command verifies the PIN before accessing the UIM contents.

This command is deprecated. QMI_UIM_VERIFY_PIN is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.14 QMI_DMS_UIM_UNBLOCK_PIN

Unblocks a blocked PIN. (Deprecated)

DMS message ID

0x0029

Version introduced

Major - 1, Minor - 1

3.14.1 Request - QMI_DMS_UIM_UNBLOCK_PIN_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
PIN Unblock Information	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	PIN Unblock Information
Length	Var			2	
Value	→	enum8	unblock_pin_id	1	Specifies the ID of the PIN to be unblocked. Values: • 1 – PIN1 (also called PIN) • 2 – PIN2
		uint8	puk_value_len	1	Number of sets of the following elements: • puk_value
		uint8	puk_value	Var	Specifies the PUK value (password) of the PIN to be unblocked.
		uint8	new_pin_value_len	1	Number of sets of the following elements: • new_pin_value
		uint8	new_pin_value	Var	Specifies the new PIN value (password) for the PIN to be unblocked.

Optional TLVs

None

3.14.2 Response - QMI_DMS_UIM_UNBLOCK_PIN_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Returned if the unblock operation failed.

Name	Version introduced	Version last modified
PIN Retries Status	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	PIN Retries Status
Length	2			2	
Value	→	uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NO_EFFECT	Operation had no effect
QMI_ERR_ARG_TOO_LONG	Device cannot handle the length of the PIN specified in the request
QMI_ERR_INCORRECT_PIN	PIN or PUK specified in the request is incorrect
QMI_ERR_PIN_BLOCKED	PIN is blocked; an unblock operation needs to be issued
QMI_ERR_PIN_PERM_BLOCKED	PIN is permanently blocked; the SIM is unusable

QMI_ERR_UIM_NOT_INITIALIZED	PIN is not yet initialized because the SIM initialization has not finished; try the PIN operation later
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_INVALID_PINID	PIN specified in the request is invalid
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed

3.14.3 Description of QMI_DMS_UIM_UNBLOCK_PIN REQ/RESP

This command unblocks a blocked PIN using the PUK provided in the request TLV. The user must enter PUK1 for PIN1, or PUK2 for PIN2.

This command is deprecated. QMI_UIM_UNBLOCK_PIN is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.15 QMI_DMS_UIM_CHANGE_PIN

Changes the PIN value. (Deprecated)

DMS message ID

0x002A

Version introduced

Major - 1, Minor - 1

3.15.1 Request - QMI_DMS_UIM_CHANGE_PIN_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
PIN Change Information	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	PIN Change Information
Length	Var			2	
Value	→	enum8	pin_id	1	Specifies the ID of the PIN to be changed. Values: • 1 – PIN1 (also called PIN) • 2 – PIN2
		uint8	old_pin_value_len	1	Number of sets of the following elements: • old_pin_value
		uint8	old_pin_value	Var	Specifies the old PIN value (old password) of the PIN.
		uint8	new_pin_value_len	1	Number of sets of the following elements: • new_pin_value
		uint8	new_pin_value	Var	Specifies the new PIN value (new password) of the PIN.

Optional TLVs

None

3.15.2 Response - QMI_DMS_UIM_CHANGE_PIN_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Returned if the enable/disable operation failed.

Name	Version introduced	Version last modified
PIN Retries Status	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	PIN Retries Status
Length	2			2	
Value	→	uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NO_EFFECT	Operation had no effect
QMI_ERR_ARG_TOO_LONG	Device cannot handle the length of the PIN specified in the request
QMI_ERR_INCORRECT_PIN	PIN specified in the request is incorrect
QMI_ERR_PIN_BLOCKED	PIN is blocked. An unblock operation needs to be issued
QMI_ERR_PIN_PERM_BLOCKED	PIN is permanently blocked; the SIM is unusable

QMI_ERR_UIM_NOT_INITIALIZED	PIN is not yet initialized because the SIM initialization has not finished; try the PIN operation later
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_INVALID_PINID	PIN specified in the request is invalid
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed

3.15.3 Description of QMI_DMS_UIM_CHANGE_PIN REQ/RESP

This command changes the old value of a specified PIN to the new value provided in the request TLV.

This command is deprecated. QMI_UIM_CHANGE_PIN is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.16 QMI_DMS_UIM_GET_PIN_STATUS

Gets the status of a PIN. (Deprecated)

DMS message ID

0x002B

Version introduced

Major - 1, Minor - 1

3.16.1 Request - QMI_DMS_UIM_GET_PIN_STATUS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.16.2 Response - QMI_DMS_UIM_GET_PIN_STATUS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Version introduced	Version last modified
PIN 1 Status	Unknown	1.1
PIN 2 Status	Unknown	1.1

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x11			1	PIN 1 Status
Length	3			2	
Value	→	enum8	status	1	Current status of the PIN. Values: • 0 – PIN is not initialized • 1 – PIN is enabled, not verified • 2 – PIN is enabled, verified • 3 – PIN is disabled • 4 – PIN is blocked • 5 – PIN is permanently blocked • 6 – PIN is unblocked • 7 – PIN is changed
		uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.
Type	0x12			1	PIN 2 Status
Length	3			2	
Value	→	enum8	status	1	Current status of the PIN. Values: • 0 – PIN is not initialized • 1 – PIN is enabled, not verified • 2 – PIN is enabled, verified • 3 – PIN is disabled • 4 – PIN is blocked • 5 – PIN is permanently blocked • 6 – PIN is unblocked • 7 – PIN is changed
		uint8	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_UIM_NOT_INITIALIZED	PIN is not yet initialized because SIM initialization has not finished; try the PIN operation later
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed

3.16.3 Description of QMI_DMS_UIM_GET_PIN_STATUS REQ/RESP

This command returns the status of PIN1 and PIN2.

This command is deprecated. QMI_UIM_GET_CARD_STATUS is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.17 QMI_DMS_GET_DEVICE_HARDWARE_REV

Queries the hardware revision of the device.

DMS message ID

0x002C

Version introduced

Major - 1, Minor - 2

3.17.1 Request - QMI_DMS_GET_DEVICE_HARDWARE_REV_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.17.2 Response - QMI_DMS_GET_DEVICE_HARDWARE_REV_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Hardware Revision	Unknown	1.2

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Hardware Revision
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	string	hardware_rev	Var	String containing the hardware revision of the device.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.17.3 Description of QMI_DMS_GET_DEVICE_HARDWARE_REV REQ/RESP

This command queries the hardware revision of the device that returns an extension of the MSM version.

3.18 QMI_DMS_GET_OPERATING_MODE

Queries the current operating mode of the device.

DMS message ID

0x002D

Version introduced

Major - 1, Minor - 2

3.18.1 Request - QMI_DMS_GET_OPERATING_MODE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.18.2 Response - QMI_DMS_GET_OPERATING_MODE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Operating Mode	1.2	1.12

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Operating Mode
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	operating_mode	1	Selected operating mode. Values: • 0 – Online • 1 – Low power • 2 – Factory Test mode • 3 – Offline • 4 – Resetting • 5 – Shutting down • 6 – Persistent low power • 8 – Conducting network test for GSM/WCDMA

Optional TLVs

Name	Version introduced	Version last modified
Offline Reason	Unknown	1.6
Hardware-Restricted Mode	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Offline Reason
Length	2			2	
Value	→	mask16	offline_reason	2	Offline reason bitmask. All unlisted bits are reserved for future use and are ignored. Values: • 0x0001 – Host image misconfiguration • 0x0002 – PRI image misconfiguration • 0x0004 – PRI version incompatible • 0x0008 – Device memory is full, cannot copy PRI information
Type	0x11			1	Hardware-Restricted Mode
Length	1			2	
Value	→	boolean	hardware_controlled_ mode	1	Hardware-Restricted mode. Values: • 0x01 – TRUE

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.18.3 Description of QMI_DMS_GET_OPERATING_MODE REQ/RESP

This command queries the current operating mode of the device. The following operating modes are available:

- Online – Indicates that the device can acquire a system and make calls
- Low Power – Lowest power consumption state from which the device can return to Online mode; indicates that the device has temporarily disabled RF
- Persistent Low Power – Same as Low Power mode, but persists even if the device is reset
- Factory Test – Special mode for manufacturer use
- Offline – Phone has deactivated RF and partially shutdown; the device must be power cycled before it can reacquire service from this mode
- Resetting – Device is in the process of power cycling
- Shutting Down – Device is in the process of shutting down
- Device is conducting a network test for GSM/WCDMA. This mode cannot be set by clients

If the operating mode returned is Offline, an optional Offline Reason TLV is provided indicating the cause of the current state. If the offline reason is not known, the TLV is omitted.

If the current operating mode was set due to a hardware override, the optional Hardware-Restricted Mode TLV is supplied with the value set to 0x01. Otherwise this TLV is omitted.

3.19 QMI_DMS_SET_OPERATING_MODE

Sets the operating mode of the device.

DMS message ID

0x002E

Version introduced

Major - 1, Minor - 2

3.19.1 Request - QMI_DMS_SET_OPERATING_MODE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Operating Mode	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Operating Mode
Length	1			2	
Value	→	enum8	operating_mode	1	Selected operating mode. Values: • 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

Optional TLVs

None

3.19.2 Response - QMI_DMS_SET_OPERATING_MODE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_DEVICE_IN_USE	Device is in use (i.e., in a call)
QMI_ERR_INVALID_ARG	Selected operating mode is invalid
QMI_ERR_INVALID_TRANSITION	Selected operating mode transition from the current operating mode is invalid
QMI_ERR_HARDWARE_RESTRICTED	Selected operating mode is invalid with the current wireless disable setting

3.19.3 Description of QMI_DMS_SET_OPERATING_MODE REQ/RESP

This command transitions operating modes based on the current mode of the device, and the mode selected. Valid transitions include:

- 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

Only Low Power mode can be used to change the device to Low Power mode, but does not modify the Persistent Low Power mode setting. If the device is not in Persistent Low Power mode, mode-only requests change the device to Low Power mode. If the device is already in Persistent Low Power mode, mode-only requests have no effect on the current mode.

Note: When in Persistent Low Power mode, only transitions to Online or regular Low Power mode cause the board to go online. Transitions to Offline (then Reset) and Shut Down power cycle the device, but upon startup, the device remains in Persistent Low Power mode.

Specifying an operating mode that is not in the valid range for the device elicits a QMI_ERR_INVALID_ARG error.

Specifying an operating mode that results in a transition not listed above elicits a QMI_ERR_INVALID_TRANSITION error.

For devices that allow hardware-controlled operating mode, it is possible that the current operating mode is enforced due to a hardware control. Changing the current operating mode to selected modes can be restricted by this hardware control, and any such requests elicit a QMI_ERR_HARDWARE_RESTRICTED error.

3.20 QMI_DMS_GET_TIME

Queries the current time of the device.

DMS message ID

0x002F

Version introduced

Major - 1, Minor - 3

3.20.1 Request - QMI_DMS_GET_TIME_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.20.2 Response - QMI_DMS_GET_TIME_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Device Time	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Device Time
Length	8			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	opaque	time_count	6	Count of 1.25 ms that have elapsed from the start of GPS Epoch time (January 6, 1980). A 6-byte integer in little-endian format.
		enum16	time_source	2	Source of the timestamp. Values: • 0 – 32 kHz device clock • 1 – CDMA network • 2 – HDR network

Optional TLVs

Name	Version introduced	Version last modified
System Time in Milliseconds	Unknown	1.4
User Time in Milliseconds	Unknown	1.4

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	System Time in Milliseconds
Length	8			2	
Value	→	uint64	sys_time_in_ms	8	Count of system time in milliseconds that have elapsed from the start of GPS Epoch time (Jan 6, 1980).
Type	0x11			1	User Time in Milliseconds
Length	8			2	
Value	→	uint64	user_time_in_ms	8	Count of user time in milliseconds that have elapsed from the start of GPS Epoch time (Jan 6, 1980).

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.20.3 Description of QMI_DMS_GET_TIME REQ/RESP

This command queries the current time of the device and returns a count of 1.25 ms that have elapsed since Jan 6, 1980 (start of GPS Epoch time), or uptime if a valid timestamp is not available.

The source of the timestamp specifies how the timestamp was determined. If a network is active for the device, the first active time source is returned. Otherwise, if no networks are active, the 32 kHz slow-clock of the device is used.

If the slow-clock on the device has never been set, or if it was set but the device remained without power for an extended period of time, an accurate timestamp is not available. If valid time cannot be returned, the device returns a timestamp that is the uptime since the device has been powered. This semi-persistent uptime must not be used as a valid GPS timestamp.

The order of precedence for networks searched is:

- CDMA
- HDR

The System Time in Milliseconds TLV returns the count of milliseconds that have elapsed since Jan 6, 1980 (start of GPS Epoch time).

If the user time is available from the device, the User Time in Milliseconds TLV is also included in the response.

3.21 QMI_DMS_GET_PRL_VER

Queries the version of the active Preferred Roaming List (PRL) of the device.

DMS message ID

0x0030

Version introduced

Major - 1, Minor - 3

3.21.1 Request - QMI_DMS_GET_PRL_VER_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.21.2 Response - QMI_DMS_GET_PRL_VER_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
PRL Version	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	PRL Version
Length	2			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint16	prl_version	2	PRL version.

Optional TLVs

Name	Version introduced	Version last modified
PRL-Only Preference	Unknown	1.5

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	PRL-Only Preference
Length	1			2	
Value	→	boolean	prl_only	1	Values: • 0 – Unset • 1 – Set

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_INVALID_ARG	Active PRL is invalid
QMI_ERR_INFO_UNAVAILABLE	Information is not available

3.21.3 Description of QMI_DMS_GET_PRL_VER REQ/RESP

This command queries the currently active PRL version of the device. This is valid only for CDMA devices (i.e., devices containing a PRL). An optional PRL-Only Preference TLV can be included in the response with information regarding whether the device is configured to register only on networks listed in the PRL (PRL-only preference).

Requesting the version of a PRL when the active PRL is invalid elicits a QMI_ERR_INVALID_ARG error.

Requests to read the PRL version on a UIM containing multiple sessions elicit a QMI_ERR_INFO_UNAVAILABLE error. Control points can either use the QMI_UIM service or QMI_DMS_GET_CURRENT_PRL_INFO (added in version 1.9) to retrieve the PRL version when this error is returned.

3.22 QMI_DMS_GET_ACTIVATION_STATE

Queries the activation state of the device.

DMS message ID

0x0031

Version introduced

Major - 1, Minor - 6

3.22.1 Request - QMI_DMS_GET_ACTIVATION_STATE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.22.2 Response - QMI_DMS_GET_ACTIVATION_STATE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Activation State	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Activation State
Length	2			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum16	activation_state	2	Service activation state. Values: • 0x00 – Service is not activated • 0x01 –Service is activated • 0x02 – Activation is connecting - Network connection in progress for automatic activation of service • 0x03 – Activation is connected - Network connection is connected for automatic activation of service • 0x04 – OTASP security is authenticated • 0x05 – OTASP NAM is downloaded • 0x06 – OTASP MDN is downloaded • 0x07 – OTASP IMSI is downloaded • 0x08 – OTASP PRL is downloaded • 0x09 – OTASP SPC is downloaded • 0x0A – OTASP settings are committed

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device

3.22.3 Description of QMI_DMS_GET_ACTIVATION_STATE REQ/RESP

This command queries the service activation state of the device.

Note: This is only supported for 3GPP2 devices.

3.23 QMI_DMS_ACTIVATE_AUTOMATIC

Requests that the device perform automatic service activation.

DMS message ID

0x0032

Version introduced

Major - 1, Minor - 6

3.23.1 Request - QMI_DMS_ACTIVATE_AUTOMATIC_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Activation Code	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Activation Code
Length	Var			2	
Value	→	uint8	act_code_len	1	Number of sets of the following elements: • act_code
		string	act_code	Var	Activation code to be used by the default activation type for the device in ASCII format (maximum 81 bytes).

Optional TLVs

None

3.23.2 Response - QMI_DMS_ACTIVATE_AUTOMATIC_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_INVALID_ARG	Invalid parameter in the request
QMI_ERR_ARG_TOO_LONG	Activation code_len size too large
QMI_ERR_NOT_PROVISIONED	Required values are not provisioned in the device
QMI_ERR_NO_NETWORK_FOUND	Device cannot acquire service to perform the activation
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device

3.23.3 Description of QMI_DMS_ACTIVATE_AUTOMATIC REQ/RESP

This command requests that the device perform automatic service activation. Success of this command indicates that the activation procedure request has been accepted and is in progress, not that it has been completed. See QMI_DMS_EVENT_REPORT_IND (Section 3.2.4) for the indications generated as the activation state is changed automatically.

The type of automatic activation initiated by this command is determined automatically by the device. The available types of automatic activation for 3GPP2 include the following:

- Over-the-air service provisioning (OTASP) – Initiates a call to the serving network to download provisioning information. See [S7] for details on this procedure. A valid activation code must be specified for this type of request.

After activation has successfully completed, the device must be power cycled before the new activation parameters take effect.

Requesting an activation type not supported by the device elicits a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

Requesting automatic activation when the minimal provisioning information necessary to acquire service is not found elicits a QMI_ERR_NOT_PROVISIONED error.

Requesting automatic activation when the device cannot acquire service from the network elicits a QMI_ERR_NO_NETWORK_FOUND error.

Note: This is only supported for 3GPP2 devices.

3.24 QMI_DMS_ACTIVATE_MANUAL

Requests that the device perform manual service activation.

DMS message ID

0x0033

Version introduced

Major - 1, Minor - 6

3.24.1 Request - QMI_DMS_ACTIVATE_MANUAL_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Manual Activation Data	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Manual Activation Data
Length	Var			2	
Value	→	char	spc	6	Service programming code in ASCII format (digits 0 to 9 only).
		uint16	sid	2	System identification number
		uint8	mdn_len	1	Number of sets of the following elements: • mdn
		string	mdn	Var	String containing the mobile directory number (maximum 15 bytes).
		uint8	min_len	1	Number of sets of the following elements: • min
		string	min	Var	String containing the mobile identification number (maximum 15 bytes).

Optional TLVs

Name	Version introduced	Version last modified
MN-HA Key	Unknown	1.6
MN-AAA Key	Unknown	1.6
Preferred Roaming List	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x11			1	MN-HA Key
Length	Var			2	
Value	→	uint8	mn_ha_key_len	1	Number of sets of the following elements: • mn_ha_key
		string	mn_ha_key	Var	String containing the MN-HA key (maximum 16 bytes).
Type	0x12			1	MN-AAA Key
Length	Var			2	
Value	→	uint8	mn_aaa_key_len	1	Number of sets of the following elements: • mn_aaa_key
		string	mn_aaa_key	Var	String containing the MN-AAA key (maximum 16 bytes).
Type	0x13			1	Preferred Roaming List
Length	Var			2	
Value	→	uint16	prl_total_len	2	PRL total length (maximum 16384)
		uint16	prl_len	2	Number of sets of the following elements: • prl
		uint8	prl_seg_num	1	PRL segment sequence number
		uint8	prl	Var	PRL segment data

3.24.2 Response - QMI_DMS_ACTIVATE_MANUAL_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_AUTHENTICATION_FAILED	Authentication of the supplied SPC failed
QMI_ERR_AUTHENTICATION_LOCK	Maximum number of authentication failures has been reached
QMI_ERR_DEVICE_IN_USE	Device is currently being provisioned by another control point
QMI_ERR_ARG_TOO_LONG	PRL total size is too large
QMI_ERR_SEGMENT_TOO_LONG	PRL segment size is too large
QMI_ERR_SEGMENT_ORDER	PRL segment order is incorrect
QMI_ERR_INVALID_ARG	Invalid parameter in the request
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device

3.24.3 Description of QMI_DMS_ACTIVATE_MANUAL REQ/RESP

This command is a service programming request and is protected by the service programming security of QMI. The correct service programming authentication code must be specified for this command. Requests with an invalid SPC elicit a QMI_ERR_AUTHENTICATION_FAILED error. If too many requests are made with an invalid SPC by any control point, the device enters an authentication locked state and elicits a QMI_ERR_AUTHENTICATION_LOCK error. Once the authentication lock state is reached, the device automatically issues a power-down procedure and shuts down. Upon rebooting, the authentication lock state is removed and the device processes service programming requests.

For CDMA devices, this command initiates a manual activation request to activate the device with new service provisioning information. The parameters specified for manual provisioning are defined in [S7], but are user-specified instead of downloaded from the network. Manual provisioning also does not require that a device be able to acquire service to be activated.

After activation has successfully completed, the device must be power cycled before the new activation parameters take effect.

Architecture restrictions limit the maximum size of QMI transactions and prevent large PRLs from being written in one request. PRLs that are less than the maximum segment length must be provisioned using a single request with the segment size equal to the total PRL size.

PRLs over the maximum segment length must be specified as a series of requests in separate transactions.

Each request must provide the mandatory TLV along with the optional Preferred Roaming List TLV containing the total PRL size, segment size, sequence number, and the PRL data to be appended to the data already specified in the series. Once all segments have been supplied, as indicated by the sum of the segment lengths equaling the total length specified, the full PRL is validated and written to the device. Requesting a total PRL size greater than the maximum elicits a QMI_ERR_ARG_TOO_LONG error, while requests with the segment size greater than the maximum elicit a QMI_ERR_SEGMENT_TOO_LONG error. Also, requests that have a total PRL size different from previous requests elicit a QMI_ERR_INVALID_ARG error.

To ensure that segments are processed in the correct order, a sequence number is included with each PRL segment. The first request of a series specifies the sequence number as zero and the number is incremented by one for every successful segment received by the device. If a duplicate or noncontiguous sequence number is received, a QMI_ERR_SEGMENT_ORDER error is returned.

Once a control point starts a PRL provisioning sequence, requests by other control points are rejected until the control point that first initiated the sequence has finished successfully or failed. Requests during this state by other control points elicit a QMI_ERR_DEVICE_IN_USE error.

If an error is encountered or the control point exits before a PRL series is completed, the PRL update is aborted. All previous PRL segments are discarded and the device becomes available for other control points to provision. The PRL series must be restarted beginning with a zero sequence-numbered request, as previously specified.

The correct service programming authentication code must be specified in the manual activation request. Specifying an invalid service programming authentication code elicits a QMI_ERR_AUTHENTICATION_FAILED error.

Requesting manual activation from a device currently in an active state elicits a QMI_ERR_DEVICE_IN_USE error.

Error checking is performed on all specified parameters before any updates are committed to the device. Any request made with an invalid parameter result in the provisioning being aborted and elicit a QMI_ERR_INVALID_ARG error.

Note: This is only supported for 3GPP2 devices.

3.25 QMI_DMS_GET_USER_LOCK_STATE

Queries the state of the user lock maintained by the device.

DMS message ID

0x0034

Version introduced

Major - 1, Minor - 3

3.25.1 Request - QMI_DMS_GET_USER_LOCK_STATE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.25.2 Response - QMI_DMS_GET_USER_LOCK_STATE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
User Lock State	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	User Lock State
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	boolean	lock_enabled	1	Current state of the lock. Values: • 0 – Disabled • 1 – Enabled

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.25.3 Description of QMI_DMS_GET_USER_LOCK_STATE REQ/RESP

This command queries the current user lock state maintained by the device.

3.26 QMI_DMS_SET_USER_LOCK_STATE

Sets the user lock state maintained by the device.

DMS message ID

0x0035

Version introduced

Major - 1, Minor - 3

3.26.1 Request - QMI_DMS_SET_USER_LOCK_STATE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
User Lock State	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	User Lock State
Length	5			2	
Value	→	enum8	lock_state	1	Current state of the lock. Values: • 0 – Disabled • 1 – Enabled
		char	lock_code	4	4-byte code set for the lock in ASCII format (digits 0 to 9 only).

Optional TLVs

None

3.26.2 Response - QMI_DMS_SET_USER_LOCK_STATE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_INVALID_ARG	Specified lock state is invalid
QMI_ERR_AUTHENTICATION_FAILED	Specified lock code is incorrect

3.26.3 Description of QMI_DMS_SET_USER_LOCK_STATE REQ/RESP

This command sets the user lock state maintained by the device.

Requests to change the state to an invalid value elicit a QMI_ERR_INVALID_ARG error.

Requests to change the state without correctly specifying the previous lock code elicit a QMI_ERR_AUTHENTICATION_FAILED error.

3.27 QMI_DMS_SET_USER_LOCK_CODE

Sets the user lock code maintained by the device.

DMS message ID

0x0036

Version introduced

Major - 1, Minor - 3

3.27.1 Request - QMI_DMS_SET_USER_LOCK_CODE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
User Lock Code	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	User Lock Code
Length	8			2	
Value	→	char	cur_code	4	Current 4-byte code to use for the lock in ASCII format (digits 0 to 9 only).
		char	new_code	4	New 4-byte code to use for the lock in ASCII format (digits 0 to 9 only).

Optional TLVs

None

3.27.2 Response - QMI_DMS_SET_USER_LOCK_CODE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_AUTHENTICATION_ FAILED	Specified lock code is incorrect

3.27.3 Description of QMI_DMS_SET_USER_LOCK_CODE REQ/RESP

This command sets the user lock code maintained by the device.

Requests to change the code without correctly specifying the previous lock code elicit a QMI_ERR_AUTHENTICATION_FAILED error.

3.28 QMI_DMS_READ_USER_DATA

Queries the user data maintained by the device.

DMS message ID

0x0037

Version introduced

Major - 1, Minor - 6

3.28.1 Request - QMI_DMS_READ_USER_DATA_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.28.2 Response - QMI_DMS_READ_USER_DATA_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
User Data	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	User Data
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint16	data_len	2	Number of sets of the following elements: • data
		uint8	data	Var	User data from/to persistent storage (maximum 512).

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device

3.28.3 Description of QMI_DMS_READ_USER_DATA REQ/RESP

This command reads the user data maintained in persistent storage by the device. The entire contents of the user data stored on the device is returned in the response. If no data has previously been written, the User Data TLV in the response is set to zero and no data follows.

Requests to read when user data is not supported by the device elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

3.29 QMI_DMS_WRITE_USER_DATA

Writes user data maintained by the device.

DMS message ID

0x0038

Version introduced

Major - 1, Minor - 6

3.29.1 Request - QMI_DMS_WRITE_USER_DATA_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
User Data	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	User Data
Length	Var			2	
Value	→	uint16	data_len	2	Number of sets of the following elements: • data
		uint8	data	Var	User data from/to persistent storage (maximum 512).

Optional TLVs

None

3.29.2 Response - QMI_DMS_WRITE_USER_DATA_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device
QMI_ERR_INVALID_ARG	Invalid parameter in the request

3.29.3 Description of QMI_DMS_WRITE_USER_DATA REQ/RESP

This command writes the user data maintained in persistent storage by the device. Once data is written it can subsequently be read using QMI_DMS_READ_USER_DATA (Section 3.28.3).

Requests to write when user data is not supported elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

Requesting that the device write more than the maximum size limit elicits a QMI_ERR_INVALID_ARG error.

3.30 QMI_DMS_READ_ERI_FILE

Queries the Extended Roaming Indicator (ERI) file stored on the device.

DMS message ID

0x0039

Version introduced

Major - 1, Minor - 6

3.30.1 Request - QMI_DMS_READ_ERI_FILE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.30.2 Response - QMI_DMS_READ_ERI_FILE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
ERI File	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	ERI File
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint16	eri_data_len	2	Number of sets of the following elements: • eri_data
		uint8	eri_data	Var	ERI data read from persistent storage (maximum 1024).

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NOT_PROVISIONED	Device has not written an ERI file
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation not supported by the device

3.30.3 Description of QMI_DMS_READ_ERI_FILE REQ/RESP

This command reads the ERI file maintained in persistent storage by the device. The entire contents of the stored ERI file are returned in the response.

Requests to read an ERI file on a device not provisioned with this file elicit a QMI_ERR_NOT_PROVISIONED error.

Requests to read an ERI file on a device that does not support this feature elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

3.31 QMI_DMS_RESTORE_FACTORY_DEFAULTS

Requests that the device reset all settings to factory defined values.

DMS message ID

0x003A

Version introduced

Major - 1, Minor - 6

3.31.1 Request - QMI_DMS_RESTORE_FACTORY_DEFAULTS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Service Programming Code	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Service Programming Code
Length	6			2	
Value	→	char	spc	6	Service programming code in ASCII format (digits 0 to 9 only).

Optional TLVs

None

3.31.2 Response - QMI_DMS_RESTORE_FACTORY_DEFAULTS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_AUTHENTICATION_FAILED	Authentication of supplied SPC failed
QMI_ERR_AUTHENTICATION_LOCK	Maximum number of authentication failures has been reached
QMI_ERR_INVALID_ARG	SPC contains one or more invalid values
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device

3.31.3 Description of QMI_DMS_RESTORE_FACTORY_DEFAULTS REQ/RESP

This command is a service programming request and is protected by the service programming security of QMI. Only the SPC, not a one-time-subsidy-lock code, can be used to restore the factory default settings of the device. The correct service programming authentication code must be specified for this command. Requests with an invalid SPC elicit a QMI_ERR_AUTHENTICATION_FAILED error. If too many requests are made with an invalid SPC by any control point, the device enters an authentication locked state and elicits a QMI_ERR_AUTHENTICATION_LOCK error. Once the authentication lock state is reached, the device automatically issues a power-down procedure and shuts down. Upon rebooting, the authentication lock state is removed and the device processes service programming requests.

This command resets the factory provisioned settings of the device and results in the removal of all user subscription information. The device must be power cycled before the reset settings take effect. After the device has been restored, new user account information must be provisioned before the device can be used again.

Error checking is performed on all specified parameters before any updates are committed to the device. Any request made with an invalid parameter results in the provisioning aborting and elicits a QMI_ERR_INVALID_ARG error.

Note that 3GPP devices that do not use an SPC must specify six zeros in this request.

3.32 QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE

Requests the device to validate a specified service programming code.

DMS message ID

0x003B

Version introduced

Major - 1, Minor - 3

3.32.1 Request - QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Programming Code	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Programming Code
Length	6			2	
Value	→	char	spc	6	Service programming code in ASCII format (digits 0 to 9 only).

Optional TLVs

None

3.32.2 Response - QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_AUTHENTICATION_FAILED	Authentication of supplied SPC failed
QMI_ERR_AUTHENTICATION_LOCK	Maximum number of authentication failures has been reached

3.32.3 Description of QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE REQ/RESP

This command is a service programming request and is protected by the service programming security of QMI. Only the SPC, not the one-time-subsidy-lock code, can be used to restore the factory default settings of the device. The correct service programming authentication code must be specified for this command. Requests with an invalid SPC elicit a QMI_ERR_AUTHENTICATION_FAILED error. If too many requests with an invalid SPC are made by any control point, the device enters an authentication locked state and elicits a QMI_ERR_AUTHENTICATION_LOCK error. Once the authentication lock state is reached, the device automatically issues a power-down procedure and shuts down. Upon rebooting, the authentication lock state is removed and the device processes service programming requests.

This command validates a specified SPC against the SPC provisioned for the device. No other operation is performed and the SPC is not remembered by the device. The SPC must still be provided in future command requests, as required.

Note that 3GPP devices that do not use an SPC must specify six zeros in this request.

3.33 QMI_DMS_UIM_GET_ICCID

Queries the Integrated Circuit Card ID (ICCID) of the UIM for the device. (Deprecated)

DMS message ID

0x003C

Version introduced

Major - 1, Minor - 3

3.33.1 Request - QMI_DMS_UIM_GET_ICCID_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.33.2 Response - QMI_DMS_UIM_GET_ICCID_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
UIM ICCID	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	UIM ICCID
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	string	uim_id	Var	String containing the UIM ICCID.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support UIM
QMI_ERR_UIM_NOT_INITIALIZED	ID is not yet initialized because the UIM initialization has not finished; try UIM operation later

3.33.3 Description of QMI_DMS_UIM_GET_ICCID REQ/RESP

This command queries the UIM ICCID for the device if a UIM is present.

This command is deprecated. QMI_UIM_READ_TRANSPARENT (EF ID: 3F00 2FE2) is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.34 QMI_DMS_UIM_GET_CK_STATUS

Queries the status of a UIM facility control key. (Deprecated)

DMS message ID

0x0040

Version introduced

Major - 1, Minor - 3

3.34.1 Request - QMI_DMS_UIM_GET_CK_STATUS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
UIM Personalization Facility	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	UIM Personalization Facility
Length	1			2	
Value	→	enum8	facility	1	MT or network facility (corresponding AT+CLCK value). Values: • 0 – Network personalization (PN) • 1 – Network subset personalization (PU) • 2 – Service provider personalization (PP) • 3 – Corporate personalization (PC) • 4 – UIM personalization (PF)

Optional TLVs

None

3.34.2 Response - QMI_DMS_UIM_GET_CK_STATUS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Facility CK Status	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Facility CK Status
Length	3			2	
Value	→	enum8	facility_state	1	UIM facility state. Values: • 0 – Deactivated • 1 – Activated • 2 – Blocked
		uint8	verify_reties_left	1	Indicates the number of retries left, after which the CK is blocked.
		uint8	unblock_retries_left	1	Number of unblock retries left, after which the CK is permanently blocked.

Optional TLVs

Name	Version introduced	Version last modified
Operation Blocking Facility	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Operation Blocking Facility
Length	1			2	
Value	→	uint8	operation_blocking	1	Presence of this TLV indicates that this facility is currently blocking normal operation of the device. This value can be returned only if the facility_state is not 0 (deactivated). **Note:** This value is set to 1 when the TLV is provided.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_UIM_NOT_INITIALIZED	SIM initialization has not finished; try the facility operation later
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed
QMI_ERR_INVALID_ARG	Invalid parameter in the request

3.34.3 Description of QMI_DMS_UIM_GET_CK_STATUS REQ/RESP

This command queries the status of a UIM facility Control Key (CK) for the device. The AT command equivalent to this command is AT+CLCK (see [S1]).

One or more personalization facilities can be activated on a device, as defined in Section 4 of [S6]. The granularity of each facility level must be contained within the personalization facilities that contain each level as a subset (i.e., if PU and PN are both activated on a device, the data set for PU must be contained within the scope of PN).

Activation of each personalization facility restricts the operation of the device to the specific facility data provisioned. This data is checked each time the device is powered up, as defined in Section 4 of [S6], and the device enters a limited service state if any one of the personalization facility checks fails. If a queried personalization is activated, but this check fails, the optional Operation Blocking Facility TLV is included to indicate that the facility must be deactivated with the proper control key before an operation can be supported.

The personalization facilities can be activated in a number of combinations, but the order in which they are checked by the device is based on the granularity of each facility. This order of precedence is defined in Table 1 of [S6]. When multiple facilities are activated, only one can be blocking an operation at a time. When this is the case, that facility must be unlocked before the status of the facilities that are a subset of the currently blocking facility are checked (i.e., PN must be deactivated if it is blocking operation, before it can be determined whether or not PU blocks operation).

If a personalization facility is blocking normal operation and verify_retries_left is zero, the unblock CK must be provided to deactivate the facility.

Requests to query the status when the message is not supported by the device elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

This command is deprecated. QMI_UIM_GET_CONFIGURATION is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.35 QMI_DMS_UIM_SET_CK_PROTECTION

Sets the protection of a UIM facility control key. (Deprecated)

DMS message ID

0x0041

Version introduced

Major - 1, Minor - 3

3.35.1 Request - QMI_DMS_UIM_SET_CK_PROTECTION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
UIM Personalization Facility	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	UIM Personalization Facility
Length	Var			2	
Value	→	enum8	facility	1	UIM Personalization facility (corresponding AT+CLCK value). Values: • 0 – Network personalization (PN) • 1 – Network subset personalization (PU) • 2 – Service provider personalization (PP) • 3 – Corporate personalization (PC) • 4 – UIM personalization (PF)
		enum8	facility_state	1	UIM facility state. Values: • 0 – Deactivated
		uint8	facility_ck_len	1	Number of sets of the following elements: • facility_ck
		string	facility_ck	Var	Facility depersonalization control key string in ASCII text (maximum 8 bytes).

Optional TLVs

None

3.35.2 Response - QMI_DMS_UIM_SET_CK_PROTECTION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Version introduced	Version last modified
Facility CK Retry Status	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Facility CK Retry Status
Length	1			2	
Value	→	uint8	verify_retries_left	1	Number of retries left, after which the CK is blocked.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_UIM_NOT_INITIALIZED	SIM initialization has not finished; try the facility operation later
QMI_ERR_INVALID_ARG	Invalid parameter in the request
QMI_ERR_ARG_TOO_LONG	Device cannot handle the length of the specified CK
QMI_ERR_NO_EFFECT	Requested state was already set for the facility
QMI_ERR_INCORRECT_PIN	CK specified in the request is incorrect
QMI_ERR_PIN_BLOCKED	CK is blocked; an unblock operation needs to be issued
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device

3.35.3 Description of QMI_DMS_UIM_SET_CK_PROTECTION REQ/RESP

This command sets the state of a UIM facility control key for the device. The AT command equivalent to this command is AT+CLCK (refer to [S1]).

Requests to set the state when the message is not supported by the device elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

This command is deprecated. QMI_UIM_DEPERSONALIZATION is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.36 QMI_DMS_UIM_UNBLOCK_CK

Unblocks a UIM facility control key. (Deprecated)

DMS message ID

0x0042

Version introduced

Major - 1, Minor - 3

3.36.1 Request - QMI_DMS_UIM_UNBLOCK_CK_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
UIM Personalization Facility	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	UIM Personalization Facility
Length	Var			2	
Value	→	enum8	facility	1	UIM personalization facility (corresponding AT+CLCK value). Values: • 0 – Network personalization (PN) • 1 – Network subset personalization (PU) • 2 – Service provider personalization (PP) • 3 – Corporate personalization (PC) • 4 – UIM personalization (PF)
		uint8	facility_unblock_ck_len	1	Number of sets of the following elements: • facility_unblock_ck
		string	facility_unblock_ck	Var	Facility control key string in ASCII text (maximum 8 bytes).

Optional TLVs

None

3.36.2 Response - QMI_DMS_UIM_UNBLOCK_CK_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Version introduced	Version last modified
Facility CK Retry Status	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Facility CK Retry Status
Length	1			2	
Value	→	uint8	unblock_retries_left	1	Number of unblock retries left, after which the CK is permanently blocked.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_UIM_NOT_INITIALIZED	SIM initialization has not finished; try the facility operation later
QMI_ERR_INVALID_ARG	Invalid parameter in the request
QMI_ERR_ARG_TOO_LONG	Device cannot handle the length of the specified CK
QMI_ERR_NO_EFFECT	Requested state was already set for the facility
QMI_ERR_INCORRECT_PIN	Unblock CK specified in the request is incorrect
QMI_ERR_PIN_PERM_BLOCKED	Unblock CK is permanently blocked
QMI_ERR_ACCESS_DENIED	Operation cannot be performed because the UIM cannot be accessed
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device

3.36.3 Description of QMI_DMS_UIM_UNBLOCK_CK REQ/RESP

This command unblocks a UIM facility control key for the device.

Requests to set the state when the message is not supported by the device elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

This command is deprecated. QMI_UIM_DEPERSONALIZATION is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.37 QMI_DMS_UIM_GET_IMSI

Queries the International Mobile Station Identity (IMSI) of the UIM for the device. (Deprecated)

DMS message ID

0x0043

Version introduced

Major - 1, Minor - 3

3.37.1 Request - QMI_DMS_UIM_GET_IMSI_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.37.2 Response - QMI_DMS_UIM_GET_IMSI_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
International Mobile Subscriber ID	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	International Mobile Subscriber ID
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	string	imsi	Var	String containing the international mobile subscriber ID.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support UIM
QMI_ERR_UIM_NOT_INITIALIZED	ID is not yet initialized, because the UIM initialization has not finished; try the UIM operation later

3.37.3 Description of QMI_DMS_UIM_GET_IMSI REQ/RESP

This command queries the IMSI for the device, if it is available in the device provisioning.

This command is deprecated. QMI_UIM_READ_TRANSPARENT (EF ID: 3F00 7F20 6F07 for 2G card and 3F00 7FFF 6F07 for 3G card) is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.38 QMI_DMS_UIM_GET_STATE

Queries the state of the UIM. (Deprecated)

DMS message ID

0x0044

Version introduced

Major - 1, Minor - 3

3.38.1 Request - QMI_DMS_UIM_GET_STATE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.38.2 Response - QMI_DMS_UIM_GET_STATE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
UIM State	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	UIM State
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	uim_state	1	UIM state. Values: • 0x00 – UIM initialization completed • 0x01 – UIM is locked or the UIM failed • 0x02 – UIM is not present • 0x03 – Reserved • 0xFF – UIM state is currently unavailable

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device

3.38.3 Description of QMI_DMS_UIM_GET_STATE REQ/RESP

This command queries the state of the UIM for the device.

Requests to query the state when the message is not supported by the device elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

This command is deprecated. QMI_UIM_GET_CARD_STATUS is the equivalent command in the QMI_UIM service and should be used in its place (refer to [Q4]).

3.39 QMI_DMS_GET_BAND_CAPABILITY

Queries the band capability of the device.

DMS message ID

0x0045

Version introduced

Major - 1, Minor - 3

3.39.1 Request - QMI_DMS_GET_BAND_CAPABILITY_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.39.2 Response - QMI_DMS_GET_BAND_CAPABILITY_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response. This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
Band Capability	Unknown	1.3

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Band Capability
Length	8			2	
Value	→	mask	band_capability	8	Bitmask of bands supported by the device; see Appendix A for the definition of these values.

Optional TLVs

Name	Version introduced	Version last modified
LTE Band Capability	Unknown	1.21
TDS Band Capability	1.10	1.10

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	LTE Band Capability
Length	8			2	
Value	→	mask	lte_band_capability	8	This TLV is present on devices that support LTE bands. Bitmask of LTE bands supported by the device; see Appendix B for the definition of these values.
Type	0x11			1	TDS Band Capability
Length	8			2	
Value	→	mask	tds_band_capability	8	This TLV is present on devices that support TDS bands. Bitmask of TDS bands supported by the device. Values: • 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

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.39.3 Description of QMI_DMS_GET_BAND_CAPABILITY REQ/RESP

This command queries the band capability of the device.

3.40 QMI_DMS_GET_FACTORY_SKU

Queries the factory provisioned Stock Keeping Unit (SKU).

DMS message ID

0x0046

Version introduced

Major - 1, Minor - 6

3.40.1 Request - QMI_DMS_GET_FACTORY_SKU_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.40.2 Response - QMI_DMS_GET_FACTORY_SKU_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Factory SKU	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Factory SKU
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	string	factory_serial_number	Var	Factory serial number string in ASCII format (maximum 128 bytes).

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device

3.40.3 Description of QMI_DMS_GET_FACTORY_SKU REQ/RESP

This command queries the SKU provisioned in the device by the factory.

Requests to query the serial number when no value has been provisioned elicit a QMI_ERR_NOT_PROVISIONED error.

3.41 QMI_DMS_SET_TIME

Sets the time on the device.

DMS message ID

0x004B

Version introduced

Major - 1, Minor - 4

3.41.1 Request - QMI_DMS_SET_TIME_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Time	Unknown	1.4

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Time
Length	8			2	
Value	→	uint64	time_in_ms	8	Count of time in milliseconds that have elapsed from the start of GPS Epoch time (Jan 6, 1980).

Optional TLVs

Name	Version introduced	Version last modified
Time Reference Type	Unknown	1.4

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Time Reference Type
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum	time_reference_type	4	Time reference used while setting the time. Values: • 0x00000000 – User time • 0x00000001 to 0xFFFFFFFF – Reserved for future extension

3.41.2 Response - QMI_DMS_SET_TIME_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.41.3 Description of QMI_DMS_SET_TIME REQ/RESP

This command is used to set the time on the device. The optional Time Reference Type TLV allows the control point to choose the type of time reference for different notions of time supported in the system. If this TLV is omitted from the request, the QMI_DMS service assumes that the request is to set the user time (0x00000000). Requests containing a time_reference_type value that is not recognized by the service elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

3.42 QMI_DMS_GET_ALT_NET_CONFIG

Queries the alternative network interface configuration used for the device.

DMS message ID

0x004D

Version introduced

Major - 1, Minor - 6

3.42.1 Request - QMI_DMS_GET_ALT_NET_CONFIG_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.42.2 Response - QMI_DMS_GET_ALT_NET_CONFIG_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Alternative Net Configuration	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Alternative Net Configuration
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	boolean	alt_net_config	1	Alternative network interface configuration. If not provisioned, the Disabled setting is used by the device as default. Values: • 0 – Disabled • 1 – Enabled

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NOT_PROVISIONED	Preference value is not set
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device

3.42.3 Description of QMI_DMS_GET_ALT_NET_CONFIG REQ/RESP

This command queries the network interface configuration setting used by the device. This setting specifies whether the device enumerates the RmNet interface(s) using the standard or alternate interface number(s).

Requests to query the configuration when the value is not set on the device elicit a QMI_ERR_NOT_PROVISIONED error.

Requests to query the configuration when the message is not supported by the device elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

3.43 QMI_DMS_SET_ALT_NET_CONFIG

Sets the alternative network interface configuration used for the device.

DMS message ID

0x004E

Version introduced

Major - 1, Minor - 6

3.43.1 Request - QMI_DMS_SET_ALT_NET_CONFIG_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Alternative Net Configuration	Unknown	1.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Alternative Net Configuration
Length	1			2	
Value	→	boolean	alt_net_config	1	Alternative network interface configuration. Values: • 0 – Disabled • 1 – Enabled

Optional TLVs

None

3.43.2 Response - QMI_DMS_SET_ALT_NET_CONFIG_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device
QMI_ERR_NO_EFFECT	Specified value is already set

3.43.3 Description of QMI_DMS_SET_ALT_NET_CONFIG REQ/RESP

This command sets the network interface configuration setting used by the device. This setting specifies whether the device enumerates the RmNet interface(s) using the standard or alternate number(s).

The device and host system must both support the alternate interface number(s) used by the device. Successfully changing the device setting without having a host driver with similar support can cause the RmNet interface(s) to not function properly. Clients must not change this setting if the host driver support is uncertain.

After the configuration has been successfully changed, the device must be power cycled before the new settings take effect. This must be done immediately following the message returning successfully, and before any power state changes are done on the host system.

Requests to set the configuration to the current setting elicit a QMI_ERR_NO_EFFECT error.

Requests to set the configuration when the message is not supported by the device in its current configuration elicit a QMI_ERR_OP_DEVICE_UNSUPPORTED error.

3.44 QMI_DMS_GET_SW_VERSION

Queries the software version from the device.

DMS message ID

0x0051

Version introduced

Major - 1, Minor - 5

3.44.1 Request - QMI_DMS_GET_SW_VERSION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.44.2 Response - QMI_DMS_GET_SW_VERSION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Name	Version introduced	Version last modified
Software Version Information	Unknown	1.5

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Software Version Information
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	string	sw_version	Var	String representing the software version information.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_NOT_PROVISIONED	Device could not retrieve the requested data because there is no provision in the device.

3.44.3 Description of QMI_DMS_GET_SW_VERSION REQ/RESP

This command queries the software version information of the device, which is represented by a string of characters.

3.45 QMI_DMS_SET_SPC

Changes the service programming code of the device after authentication.

DMS message ID

0x0052

Version introduced

Major - 1, Minor - 5

3.45.1 Request - QMI_DMS_SET_SPC_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Current SPC	Unknown	1.5
New SPC	Unknown	1.5

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Current SPC
Length	6			2	
Value	→	char	curr_spc	6	SPC for authentication in ASCII format (digits 0 to 9 only).
Type	0x02			1	New SPC
Length	6			2	
Value	→	char	new_spc	6	New SPC in ASCII format (digits 0 to 9 only).

Optional TLVs

None

3.45.2 Response - QMI_DMS_SET_SPC_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_AUTHENTICATION_ FAILED	Authentication of supplied SPC failed
QMI_ERR_AUTHENTICATION_LOCK	Maximum number of authentication failures has been reached

3.45.3 Description of QMI_DMS_SET_SPC REQ/RESP

This command changes the service programming code of the device. There are two TLVs required in the request: Current SPC and the New SPC. The new SPC takes effect after authentication and validation of the current SPC.

3.46 QMI_DMS_GET_CURRENT_PRL_INFO

Queries the currently active PRL information of the device.

DMS message ID

0x0053

Version introduced

Major - 1, Minor - 9

3.46.1 Request - QMI_DMS_GET_CURRENT_PRL_INFO_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.46.2 Response - QMI_DMS_GET_CURRENT_PRL_INFO_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Version introduced	Version last modified
PRL Version	Unknown	1.9
PRL Only Preference	Unknown	1.9

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	PRL Version
Length	2			2	
Value	→	uint16	prl_version	2	PRL version
Type	0x11			1	PRL Only Preference
Length	1			2	
Value	→	boolean	prl_only	1	Values: • 0 – Unset • 1 – Set

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_INFO_UNAVAILABLE	PRL has not been loaded onto device
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Operation is not supported by the device

3.46.3 Description of QMI_DMS_GET_CURRENT_PRL_INFO REQ/RESP

This command queries the currently active PRL information of the device. This is valid only for CDMA devices (devices containing a PRL). This command returns the PRL that is loaded to the device and can change during device power-up or operating mode change. Control points can register for the PRL Init Reporting event using QMI_DMS_SET_EVENT_REPORT and send this command again to query new values when the PRL Init Notification indication is received.

3.47 QMI_DMS_BIND_SUBSCRIPTION

Associates the requesting control point with the requested subscription.

DMS message ID

0x0054

Version introduced

Major - 1, Minor - 15

3.47.1 Request - QMI_DMS_BIND_SUBSCRIPTION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Bind Subscription	1.15	1.19

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Bind Subscription
Length	4			2	
Value	→	enum	bind_subs	4	Subscription to which to bind. Values: • DMS_PRIMARY_SUBS (0x0001) – Primary • DMS_SECONDARY_SUBS (0x0002) – Secondary • DMS_TERTIARY_SUBS (0x0003) – Tertiary

Optional TLVs

None

3.47.2 Response - QMI_DMS_BIND_SUBSCRIPTION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_MISSING_ARG	Required TLV was missing in the request
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point, or the message was corrupted during transmission
QMI_ERR_INVALID_ARG	Value field of one or more TLVs in the request message contains an invalid value

3.47.3 Description of QMI_DMS_BIND_SUBSCRIPTION REQ/RESP

This command binds the control point to the requested subscription, after which control point requests/indications are acted upon/sent to the subscription it is bound to. If this message is not sent by the control point, it is bound to the default (primary) subscription to which it is bound. This command is client-specific, so each control point must send this message to change subscription binding.

3.48 QMI_DMS_GET_BIND_SUBSCRIPTION

Queries the subscription associated with the control point.

DMS message ID

0x0055

Version introduced

Major - 1, Minor - 15

3.48.1 Request - QMI_DMS_GET_BIND_SUBSCRIPTION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.48.2 Response - QMI_DMS_GET_BIND_SUBSCRIPTION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Version introduced	Version last modified
Bound Subscription	1.15	1.19

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Bound Subscription
Length	4			2	
Value	→	enum	bind_subscription	4	Values: • DMS_PRIMARY_SUBS (0x0001) – Primary • DMS_SECONDARY_SUBS (0x0002) – Secondary • DMS_TERTIARY_SUBS (0x0003) – Tertiary

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.48.3 Description of QMI_DMS_GET_BIND_SUBSCRIPTION REQ/RESP

This command queries the current subscription this control point is bound to. If no bind_subscription message was previously sent by this control point, the primary subscription (default) is returned.

3.49 QMI_DMS_SET_AP_SW_VERSION

Sets the AP software version on the modem required for an Auto Register Short message.

DMS message ID

0x0056

Version introduced

Major - 1, Minor - 27

3.49.1 Request - QMI_DMS_SET_AP_SW_VERSION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
AP Software Version	1.27	1.27

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	AP Software Version
Length	Var			2	
Value	→	string	ap_sw_version	Var	String representing the AP software version information.

Optional TLVs

None

3.49.2 Response - QMI_DMS_SET_AP_SW_VERSION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_MALFORMED_MSG	The message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_INVALID_ARG	Invalid parameter in the request

3.49.3 Description of QMI_DMS_SET_AP_SW_VERSION REQ/RESP

This command sets the AP software version on the modem, which is required for the modem to send an Auto Register Short message on bootup. Note that this command does not affect or override values returned by QMI_DMS_GET_SW_VERSION.

3.50 QMI_DMS_GET_CDMA_LOCK_MODE

Requests the CDMA Lock mode status.

DMS message ID

0x0057

Version introduced

Major - 1, Minor - 24

3.50.1 Request - QMI_DMS_GET_CDMA_LOCK_MODE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.50.2 Response - QMI_DMS_GET_CDMA_LOCK_MODE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

This TLV is present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
CDMA Lock Mode	1.24	1.24

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	CDMA Lock Mode
Length	4			2	
Value	→	enum	cdma_lock_mode_status	4	CDMA Lock mode status. Values: • DMS_CDMA_LOCK_MODE_OFF (0) – Phone is not CDMA locked • DMS_CDMA_LOCK_MODE_ON (1) – Phone is CDMA locked

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.50.3 Description of QMI_DMS_GET_CDMA_LOCK_MODE REQ/RESP

This command obtains information regarding the CDMA Lock mode status of the device. The information returned is described in the optional TLV table above.

3.51 QMI_DMS_SET_TEST_CONFIG

Sets the configuration type used while testing.

DMS message ID

0x0058

Version introduced

Major - 1, Minor - 25

3.51.1 Request - QMI_DMS_SET_TEST_CONFIG_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
TDS CDMA Configuration	1.25	1.26

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	TDS CDMA Configuration
Length	4			2	
Value	→	enum	tds_config	4	Configuration parameters to be used for TDS CDMA. Values: • DMS_TEST_CONFIG_TDS_PRODUCTION (0) – Use the configuration applicable in production (in the field) • DMS_TEST_CONFIG_TDS_LAB (1) – Use the configuration applicable in the lab • DMS_TEST_CONFIG_TDS_USER (2) – Use the user-defined configuration

3.51.2 Response - QMI_DMS_SET_TEST_CONFIG_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_MALFORMED_MSG	The message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_MISSING_ARG	One or more required TLVs were missing in the request
QMI_ERR_INVALID_ARG	Invalid parameter in the request

3.51.3 Description of QMI_DMS_SET_TEST_CONFIG REQ/RESP

This command is used by the control point to set the configuration parameters to be used by the UE during testing. The value set using this command is persistent. The configuration change takes effect only after a power cycle. By default the device uses DMS_TEST_CONFIG_TDS_PRODUCTION, which is the production configuration.

Each optional TLV in the request message can be sent independently to enable a test option that forces the modem to comply with some specific operator or acceptance tests that are otherwise not met by the production modem configuration. Each TLV clearly defines the setting that corresponds to the production modem configuration. This setting is the default value used until the control point sends a request to enable a specific test option.

3.52 QMI_DMS_GET_TEST_CONFIG

Gets the configuration type used for testing.

DMS message ID

0x0059

Version introduced

Major - 1, Minor - 25

3.52.1 Request - QMI_DMS_GET_TEST_CONFIG_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.52.2 Response - QMI_DMS_GET_TEST_CONFIG_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

Name	Version introduced	Version last modified
Active TDS CDMA Configuration	1.25	1.26
Desired TDS CDMA Configuration	1.25	1.26

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Active TDS CDMA Configuration
Length	4			2	
Value	→	enum	active_tds_config	4	Configuration parameters currently used for TDS CDMA. Values: • DMS_TEST_CONFIG_TDS_ PRODUCTION (0) – Use the configuration applicable in production (in the field) • DMS_TEST_CONFIG_TDS_LAB (1) – Use the configuration applicable in the lab • DMS_TEST_CONFIG_TDS_USER (2) – Use the user-defined configuration
Type	0x11			1	Desired TDS CDMA Configuration
Length	4			2	
Value	→	enum	desired_tds_config	4	Configuration parameters for TDS CDMA that were set using the last QMI_DMS_SET_TEST_CONFIG command. Values: • DMS_TEST_CONFIG_TDS_ PRODUCTION (0) – Use the configuration applicable in production (in the field) • DMS_TEST_CONFIG_TDS_LAB (1) – Use the configuration applicable in the lab • DMS_TEST_CONFIG_TDS_USER (2) – Use the user-defined configuration

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.52.3 Description of QMI_DMS_GET_TEST_CONFIG REQ/RESP

This command is used by the control point to get the configuration parameters currently in effect for testing on the UE. If the Active TDS CDMA Configuration and Desired TDS CDMA Configuration TLVs have different values, it means that the device has not been power cycled since the last QMI_DMS_SET_TEST_CONFIG command, so the change has not taken effect.

3.53 QMI_DMS_CLEAR_TEST_CONFIG

Resets the modem configuration to production values.

DMS message ID

0x005A

Version introduced

Major - 1, Minor - 26

3.53.1 Request - QMI_DMS_CLEAR_TEST_CONFIG_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.53.2 Response - QMI_DMS_CLEAR_TEST_CONFIG_RESP

Message type

Response

Sender

Service

Mandatory TLVs

The Result Code TLV (defined in Section 2.3.1) is always present in the response.

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response

3.53.3 Description of QMI_DMS_CLEAR_TEST_CONFIG REQ/RESP

This command is used by the control point to restore the modem configuration to its production configuration. The configuration change takes effect only after a power cycle. Any persistent configuration previously set using the QMI_DMS_SET_TEST_CONFIG command is reverted back to the production setting.

A Band Capability

This appendix lists the bitmasks of bands supported by the device.

Table A-1 Bands supported by device

Value	Name
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
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	QWCDMA 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

Table A-1 Bands supported by device (cont.)

Value	Name
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
Bit 60	WCDMA Japan 850 band
Bit 61	WCDMA 1500 band
Bits 62 and 63	Reserved

B LTE Band Capability

This appendix lists the bitmasks of LTE bands supported by the device.

Table B-1 LTE bands supported by device

Value	Name
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

Table B-1 LTE Bands supported by device) (cont.)

Value	Name
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

Table B-1 LTE Bands supported by device) (cont.)

Value	Name
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

C Deprecated QMI_DMS Messages

Table C-1 lists the deprecated QMI_DMS messages and their replacements.

Table C-1 Deprecated QMI_DMS messages

Message	Replacement
QMI_DMS_UIM_SET_PIN_ PROTECTION	QMI_UIM_SET_PIN_PROTECTION – Refer to [Q4].
QMI_DMS_UIM_VERIFY_PIN	QMI_UIM_VERIFY_PIN – Refer to [Q4].
QMI_DMS_UIM_UNBLOCK_PIN	QMI_UIM_UNBLOCK_PIN – Refer to [Q4].
QMI_DMS_UIM_CHANGE_PIN	QMI_UIM_CHANGE_PIN – Refer to [Q4].
QMI_DMS_UIM_GET_PIN_STATUS	QMI_UIM_GET_CARD_STATUS – Refer to [Q4].
QMI_DMS_UIM_GET_ICCID	QMI_UIM_READ_TRANSPARENT – Refer to [Q4].
QMI_DMS_UIM_GET_CK_STATUS	QMI_UIM_GET_CONFIGURATION – Refer to [Q4].
QMI_DMS_UIM_SET_CK_ PROTECTION	QMI_UIM_DEPERSONALIZATION – Refer to [Q4].
QMI_DMS_UIM_UNBLOCK_ CK	QMI_UIM_DEPERSONALIZATION – Refer to [Q4].
QMI_DMS_UIM_GET_IMSI	QMI_UIM_READ_TRANSPARENT – Refer to [Q4].
QMI_DMS_UIM_GET_STATE	QMI_UIM_GET_CARD_STATUS – Refer to [Q4].