



# *QMI Device Management Service (QMI\_DMS)*

## *Major Version 1, Minor Version 8*

### *Specification*

*80-VB816-4 K*

*November 10, 2011*

---

Submit technical questions at:

<https://support.cdmatech.com>

#### **Qualcomm Confidential and Proprietary**

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

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

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 Qualcomm confidential and proprietary information and must be shredded when discarded.

QUALCOMM is a registered trademark of QUALCOMM Incorporated in the United States and may be registered in other countries. Other product and brand names may be trademarks or registered trademarks of their respective owners. CDMA2000 is a registered certification mark of the Telecommunications Industry Association, used under license. ARM is a registered trademark of ARM Limited. QDSP is a registered trademark of QUALCOMM Incorporated in the United States and other countries.

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 Incorporated**  
**5775 Morehouse Drive**  
**San Diego, CA 92121-1714**  
**U.S.A.**

**Copyright © 2006-2011 QUALCOMM Incorporated.**  
**All rights reserved.**

# Contents

---

<b>1</b>	<b>Introduction</b>	<b>9</b>
1.1	Purpose . . . . .	9
1.2	Scope . . . . .	9
1.3	Conventions . . . . .	10
1.4	References . . . . .	10
1.5	Technical Assistance . . . . .	11
1.6	Acronyms . . . . .	11
<b>2</b>	<b>Theory of Operation</b>	<b>12</b>
2.1	Generalized QMI Service Compliance . . . . .	12
2.2	DMS Service Type . . . . .	12
2.3	Message Definition Template . . . . .	12
2.3.1	Mandatory Result TLV . . . . .	12
2.4	QMI_DMS Fundamental Concepts . . . . .	13
2.5	Service State Variables . . . . .	13
2.5.1	Shared State Variables . . . . .	13
2.5.2	State Variables Per Control Point . . . . .	13
<b>3</b>	<b>QMI_DMS Messages</b>	<b>15</b>
3.1	QMI_DMS_RESET . . . . .	18
3.1.1	Request - QMI_DMS_RESET_REQ . . . . .	18
3.1.2	Response - QMI_DMS_RESET_RESP . . . . .	18
3.1.3	Description of QMI_DMS_RESET REQ/RESP . . . . .	19
3.2	QMI_DMS_SET_EVENT_REPORT . . . . .	20
3.2.1	Request - QMI_DMS_SET_EVENT_REPORT_REQ . . . . .	20
3.2.2	Response - QMI_DMS_SET_EVENT_REPORT_RESP . . . . .	22
3.2.3	Indication - QMI_DMS_EVENT_REPORT_IND . . . . .	22
3.2.4	Description of QMI_DMS_SET_EVENT_REPORT . . . . .	25
3.3	QMI_DMS_GET_DEVICE_CAP . . . . .	27
3.3.1	Request - QMI_DMS_GET_DEVICE_CAP_REQ . . . . .	27
3.3.2	Response - QMI_DMS_GET_DEVICE_CAP_RESP . . . . .	27
3.3.3	Description of QMI_DMS_GET_DEVICE_CAP REQ/RESP . . . . .	29
3.4	QMI_DMS_GET_DEVICE_MFR . . . . .	30
3.4.1	Request - QMI_DMS_GET_DEVICE_MFR_REQ . . . . .	30
3.4.2	Response - QMI_DMS_GET_DEVICE_MFR_RESP . . . . .	30
3.4.3	Description of QMI_DMS_GET_DEVICE_MFR REQ/RESP . . . . .	31
3.5	QMI_DMS_GET_DEVICE_MODEL_ID . . . . .	32
3.5.1	Request - QMI_DMS_GET_DEVICE_MODEL_ID_REQ . . . . .	32
3.5.2	Response - QMI_DMS_GET_DEVICE_MODEL_ID_RESP . . . . .	32
3.5.3	Description of QMI_DMS_GET_DEVICE_MODEL_ID REQ/RESP . . . . .	33

3.6	QMI_DMS_GET_DEVICE_REV_ID	34
3.6.1	Request - QMI_DMS_GET_DEVICE_REV_ID_REQ	34
3.6.2	Response - QMI_DMS_GET_DEVICE_REV_ID_RESP	34
3.6.3	Description of QMI_DMS_GET_DEVICE_REV_ID REQ/RESP	35
3.7	QMI_DMS_GET_MSISDN	37
3.7.1	Request - QMI_DMS_GET_MSISDN_REQ	37
3.7.2	Response - QMI_DMS_GET_MSISDN_RESP	37
3.7.3	Description of QMI_DMS_GET_MSISDN REQ/RESP	39
3.8	QMI_DMS_GET_DEVICE_SERIAL_NUMBERS	40
3.8.1	Request - QMI_DMS_GET_DEVICE_SERIAL_NUMBERS_REQ	40
3.8.2	Response - QMI_DMS_GET_DEVICE_SERIAL_NUMBERS_RESP	40
3.8.3	Description of QMI_DMS_GET_DEVICE_SERIAL_NUMBERS REQ/RESP	41
3.9	QMI_DMS_GET_POWER_STATE	42
3.9.1	Request - QMI_DMS_GET_POWER_STATE_REQ	42
3.9.2	Response - QMI_DMS_GET_POWER_STATE_RESP	42
3.9.3	Description of QMI_DMS_GET_POWER_STATE REQ/RESP	44
3.10	QMI_DMS_UIM_SET_PIN_PROTECTION	45
3.10.1	Request - QMI_DMS_UIM_SET_PIN_PROTECTION_REQ	45
3.10.2	Response - QMI_DMS_UIM_SET_PIN_PROTECTION_RESP	46
3.10.3	Description of QMI_DMS_UIM_SET_PIN_PROTECTION REQ/RESP	47
3.11	QMI_DMS_UIM_VERIFY_PIN	48
3.11.1	Request - QMI_DMS_UIM_VERIFY_PIN_REQ	48
3.11.2	Response - QMI_DMS_UIM_VERIFY_PIN_RESP	49
3.11.3	Description of QMI_DMS_UIM_VERIFY_PIN REQ/RESP	50
3.12	QMI_DMS_UIM_UNBLOCK_PIN	51
3.12.1	Request - QMI_DMS_UIM_UNBLOCK_PIN_REQ	51
3.12.2	Response - QMI_DMS_UIM_UNBLOCK_PIN_RESP	52
3.12.3	Description of QMI_DMS_UIM_UNBLOCK_PIN REQ/RESP	53
3.13	QMI_DMS_UIM_CHANGE_PIN	54
3.13.1	Request - QMI_DMS_UIM_CHANGE_PIN_REQ	54
3.13.2	Response - QMI_DMS_UIM_CHANGE_PIN_RESP	55
3.13.3	Description of QMI_DMS_UIM_CHANGE_PIN REQ/RESP	56
3.14	QMI_DMS_UIM_GET_PIN_STATUS	57
3.14.1	Request - QMI_DMS_UIM_GET_PIN_STATUS_REQ	57
3.14.2	Response - QMI_DMS_UIM_GET_PIN_STATUS_RESP	57
3.14.3	Description of QMI_DMS_UIM_GET_PIN_STATUS REQ/RESP	59
3.15	QMI_DMS_GET_DEVICE_HARDWARE_REV	60
3.15.1	Request - QMI_DMS_GET_DEVICE_HARDWARE_REV_REQ	60
3.15.2	Response - QMI_DMS_GET_DEVICE_HARDWARE_REV_RESP	60
3.15.3	Description of QMI_DMS_GET_DEVICE_HARDWARE_REV REQ/RESP	61
3.16	QMI_DMS_GET_OPERATING_MODE	62
3.16.1	Request - QMI_DMS_GET_OPERATING_MODE_REQ	62
3.16.2	Response - QMI_DMS_GET_OPERATING_MODE_RESP	62
3.16.3	Description of QMI_DMS_GET_OPERATING_MODE REQ/RESP	64
3.17	QMI_DMS_SET_OPERATING_MODE	65
3.17.1	Request - QMI_DMS_SET_OPERATING_MODE_REQ	65
3.17.2	Response - QMI_DMS_SET_OPERATING_MODE_RESP	66
3.17.3	Description of QMI_DMS_SET_OPERATING_MODE REQ/RESP	66
3.18	QMI_DMS_GET_TIME	68
3.18.1	Request - QMI_DMS_GET_TIME_REQ	68

3.18.2	Response - QMI_DMS_GET_TIME_RESP . . . . .	68
3.18.3	Description of QMI_DMS_GET_TIME REQ/RESP . . . . .	70
3.19	QMI_DMS_GET_PRL_VER . . . . .	71
3.19.1	Request - QMI_DMS_GET_PRL_VER_REQ . . . . .	71
3.19.2	Response - QMI_DMS_GET_PRL_VER_RESP . . . . .	71
3.19.3	Description of QMI_DMS_GET_PRL_VER REQ/RESP . . . . .	73
3.20	QMI_DMS_GET_ACTIVATION_STATE . . . . .	74
3.20.1	Request - QMI_DMS_GET_ACTIVATION_STATE_REQ . . . . .	74
3.20.2	Response - QMI_DMS_GET_ACTIVATION_STATE_RESP . . . . .	74
3.20.3	Description of QMI_DMS_GET_ACTIVATION_STATE REQ/RESP . . . . .	76
3.21	QMI_DMS_ACTIVATE_AUTOMATIC . . . . .	77
3.21.1	Request - QMI_DMS_ACTIVATE_AUTOMATIC_REQ . . . . .	77
3.21.2	Response - QMI_DMS_ACTIVATE_AUTOMATIC_RESP . . . . .	78
3.21.3	Description of QMI_DMS_ACTIVATE_AUTOMATIC REQ/RESP . . . . .	78
3.22	QMI_DMS_ACTIVATE_MANUAL . . . . .	80
3.22.1	Request - QMI_DMS_ACTIVATE_MANUAL_REQ . . . . .	80
3.22.2	Response - QMI_DMS_ACTIVATE_MANUAL_RESP . . . . .	81
3.22.3	Description of QMI_DMS_ACTIVATE_MANUAL REQ/RESP . . . . .	82
3.23	QMI_DMS_GET_USER_LOCK_STATE . . . . .	84
3.23.1	Request - QMI_DMS_GET_USER_LOCK_STATE_REQ . . . . .	84
3.23.2	Response - QMI_DMS_GET_USER_LOCK_STATE_RESP . . . . .	84
3.23.3	Description of QMI_DMS_GET_USER_LOCK_STATE REQ/RESP . . . . .	86
3.24	QMI_DMS_SET_USER_LOCK_STATE . . . . .	87
3.24.1	Request - QMI_DMS_SET_USER_LOCK_STATE_REQ . . . . .	87
3.24.2	Response - QMI_DMS_SET_USER_LOCK_STATE_RESP . . . . .	88
3.24.3	Description of QMI_DMS_SET_USER_LOCK_STATE REQ/RESP . . . . .	88
3.25	QMI_DMS_SET_USER_LOCK_CODE . . . . .	89
3.25.1	Request - QMI_DMS_SET_USER_LOCK_CODE_REQ . . . . .	89
3.25.2	Response - QMI_DMS_SET_USER_LOCK_CODE_RESP . . . . .	90
3.25.3	Description of QMI_DMS_SET_USER_LOCK_CODE REQ/RESP . . . . .	90
3.26	QMI_DMS_READ_USER_DATA . . . . .	91
3.26.1	Request - QMI_DMS_READ_USER_DATA_REQ . . . . .	91
3.26.2	Response - QMI_DMS_READ_USER_DATA_RESP . . . . .	91
3.26.3	Description of QMI_DMS_READ_USER_DATA REQ/RESP . . . . .	92
3.27	QMI_DMS_WRITE_USER_DATA . . . . .	93
3.27.1	Request - QMI_DMS_WRITE_USER_DATA_REQ . . . . .	93
3.27.2	Response - QMI_DMS_WRITE_USER_DATA_RESP . . . . .	94
3.27.3	Description of QMI_DMS_WRITE_USER_DATA REQ/RESP . . . . .	94
3.28	QMI_DMS_READ_ERI_FILE . . . . .	95
3.28.1	Request - QMI_DMS_READ_ERI_FILE_REQ . . . . .	95
3.28.2	Response - QMI_DMS_READ_ERI_FILE_RESP . . . . .	95
3.28.3	Description of QMI_DMS_READ_ERI_FILE REQ/RESP . . . . .	96
3.29	QMI_DMS_RESTORE_FACTORY_DEFAULTS . . . . .	97
3.29.1	Request - QMI_DMS_RESTORE_FACTORY_DEFAULTS_REQ . . . . .	97
3.29.2	Response - QMI_DMS_RESTORE_FACTORY_DEFAULTS_RESP . . . . .	97
3.29.3	Description of QMI_DMS_RESTORE_FACTORY_DEFAULTS REQ/RESP . . . . .	98
3.30	QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE . . . . .	99
3.30.1	Request - QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE_REQ . . . . .	99
3.30.2	Response - QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE_RESP . . . . .	99

3.30.3	Description of QMI_DMS_VALIDATE_SERVICE_PROGRAMMING_CODE REQ/RESP	100
3.31	QMI_DMS_UIM_GET_ICCID	101
3.31.1	Request - QMI_DMS_UIM_GET_ICCID_REQ	101
3.31.2	Response - QMI_DMS_UIM_GET_ICCID_RESP	101
3.31.3	Description of QMI_DMS_UIM_GET_ICCID REQ/RESP	102
3.32	QMI_DMS_UIM_GET_CK_STATUS	103
3.32.1	Request - QMI_DMS_UIM_GET_CK_STATUS_REQ	103
3.32.2	Response - QMI_DMS_UIM_GET_CK_STATUS_RESP	104
3.32.3	Description of QMI_DMS_UIM_GET_CK_STATUS REQ/RESP	105
3.33	QMI_DMS_UIM_SET_CK_PROTECTION	106
3.33.1	Request - QMI_DMS_UIM_SET_CK_PROTECTION_REQ	106
3.33.2	Response - QMI_DMS_UIM_SET_CK_PROTECTION_RESP	107
3.33.3	Description of QMI_DMS_UIM_SET_CK_PROTECTION REQ/RESP	108
3.34	QMI_DMS_UIM_UNBLOCK_CK	109
3.34.1	Request - QMI_DMS_UIM_UNBLOCK_CK_REQ	109
3.34.2	Response - QMI_DMS_UIM_UNBLOCK_CK_RESP	110
3.34.3	Description of QMI_DMS_UIM_UNBLOCK_CK REQ/RESP	111
3.35	QMI_DMS_UIM_GET_IMSI	112
3.35.1	Request - QMI_DMS_UIM_GET_IMSI_REQ	112
3.35.2	Response - QMI_DMS_UIM_GET_IMSI_RESP	112
3.35.3	Description of QMI_DMS_UIM_GET_IMSI REQ/RESP	114
3.36	QMI_DMS_UIM_GET_STATE	115
3.36.1	Request - QMI_DMS_UIM_GET_STATE_REQ	115
3.36.2	Response - QMI_DMS_UIM_GET_STATE_RESP	115
3.36.3	Description of QMI_DMS_UIM_GET_STATE REQ/RESP	116
3.37	QMI_DMS_GET_BAND_CAPABILITY	117
3.37.1	Request - QMI_DMS_GET_BAND_CAPABILITY_REQ	117
3.37.2	Response - QMI_DMS_GET_BAND_CAPABILITY_RESP	117
3.37.3	Description of QMI_DMS_GET_BAND_CAPABILITY REQ/RESP	118
3.38	QMI_DMS_GET_FACTORY_SKU	119
3.38.1	Request - QMI_DMS_GET_FACTORY_SKU_REQ	119
3.38.2	Response - QMI_DMS_GET_FACTORY_SKU_RESP	119
3.38.3	Description of QMI_DMS_GET_FACTORY_SKU REQ/RESP	121
3.39	QMI_DMS_SET_TIME	122
3.39.1	Request - QMI_DMS_SET_TIME_REQ	122
3.39.2	Response - QMI_DMS_SET_TIME_RESP	123
3.39.3	Description of QMI_DMS_SET_TIME REQ/RESP	123
3.40	QMI_DMS_GET_ALT_NET_CONFIG	124
3.40.1	Request - QMI_DMS_GET_ALT_NET_CONFIG_REQ	124
3.40.2	Response - QMI_DMS_GET_ALT_NET_CONFIG_RESP	124
3.40.3	Description of QMI_DMS_GET_ALT_NET_CONFIG REQ/RESP	125
3.41	QMI_DMS_SET_ALT_NET_CONFIG	126
3.41.1	Request - QMI_DMS_SET_ALT_NET_CONFIG_REQ	126
3.41.2	Response - QMI_DMS_SET_ALT_NET_CONFIG_RESP	127
3.41.3	Description of QMI_DMS_SET_ALT_NET_CONFIG REQ/RESP	127
3.42	QMI_DMS_GET_SW_VERSION	128
3.42.1	Request - QMI_DMS_GET_SW_VERSION_REQ	128
3.42.2	Response - QMI_DMS_GET_SW_VERSION_RESP	128
3.42.3	Description of QMI_DMS_GET_SW_VERSION REQ/RESP	129

3.43	QMI_DMS_SET_SPC	130
3.43.1	Request - QMI_DMS_SET_SPC_REQ	130
3.43.2	Response - QMI_DMS_SET_SPC_RESP	131
3.43.3	Description of QMI_DMS_SET_SPC REQ/RESP	131
<b>A</b>	<b>Band Capability</b>	<b>132</b>
<b>B</b>	<b>LTE Band Capability</b>	<b>134</b>

List of Tables

1-2 Reference documents and standards . . . . . 10

1-3 Acronyms . . . . . 11

3-1 QMI\_DMS messages . . . . . 15

A-1 Bands supported by device . . . . . 132

B-1 LTE bands supported by device . . . . . 134

## 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. <ul style="list-style-type: none"> <li>Added PRL Init TLV in QMI_DMS_SET_EVENT_REPORT and QMI_DMS_EVENT_REPORT_IND</li> <li>Updated the description of QMI_DMS_GET_PRL_VER</li> <li>Added LTE Band Capability TLV in QMI_DMS_GET_BAND_CAPABILITY</li> <li>Added Appendix A and Appendix B</li> </ul>

**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<sup>®</sup> 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.

## 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

Table 1-2 lists reference documents, which may include Qualcomm documents and non-Qualcomm standards and resources. Reference documents that are no longer applicable are deleted from this table; therefore, reference numbers might not be sequential.

**Table 1-2 Reference documents and standards**

Ref.	Document	
Qualcomm		
Q1	QUALCOMM® MSM™ Interface (QMI) Architecture	80-VB816-1
Q2	Application Note: Software Glossary for Customers	CL93-V3077-1
Q3	Multimode GSDI SIM, USIM, and R-UIM Interface Specification and Operational Description	80-V5329-1
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 guide, submit a case to Qualcomm CDMA Technologies at <https://support.cdmatech.com>.

If you do not have access to the CDMATech Support Services website, register for access or send email to [support.cdmatech@qualcomm.com](mailto:support.cdmatech@qualcomm.com).

## 1.6 Acronyms

For definitions of terms and abbreviations, see [Q1]. Table 1-3 lists terms that are specific to this document.

**Table 1-3 Acronyms**

Acronym	Definition
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
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
SKU	stock keeping unit
SIM	subscriber identity module
SPC	service programming code
SVN	software version number
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 Mandatory 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 last modified
Result Code	Corresponding messages Version Introduced

Field	Field value	Parameter	Size (byte)	Description
Type	0x02		1	Result Code
Length	4		2	
Value	→	qmi_result	2	Result code <ul style="list-style-type: none"><li>• QMI_RESULT_SUCCESS</li><li>• QMI_RESULT_FAILURE</li></ul>
		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 values
report_power_state	Indicates whether a power state change is reported to the control point	<ul style="list-style-type: none"> <li>• TRUE</li> <li>• FALSE</li> </ul>	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 values
report_activation_state	Indicates whether a change in data-bearer technology is reported to the control point	<ul style="list-style-type: none"><li>• TRUE</li><li>• FALSE</li></ul>	FALSE
report_operating_mode	Indicates whether an operating mode change is reported to the control point	<ul style="list-style-type: none"><li>• TRUE</li><li>• FALSE</li></ul>	FALSE
report_uim_state	Indicates whether a UIM state change is reported to the control point	<ul style="list-style-type: none"><li>• TRUE</li><li>• FALSE</li></ul>	FALSE
report_wireless_disable	Indicates whether a wireless disable state change is reported to the control point	<ul style="list-style-type: none"><li>• TRUE</li><li>• FALSE</li></ul>	FALSE
report_prl_init	Indicates whether a PRL initialized notification is reported to the control point	<ul style="list-style-type: none"><li>• TRUE</li><li>• FALSE</li></ul>	FALSE

# 3 QMI\_DMS Messages

---

**Table 3-1 QMI\_DMS messages**

<b>Command</b>	<b>ID</b>	<b>Description</b>
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_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.
QMI_DMS_UIM_VERIFY_PIN	0x0028	Verifies the PIN before accessing the UIM contents.
QMI_DMS_UIM_UNBLOCK_PIN	0x0029	Unblocks a blocked PIN.
QMI_DMS_UIM_CHANGE_PIN	0x002A	Changes the PIN value.
QMI_DMS_UIM_GET_PIN_STATUS	0x002B	Gets the status of a PIN.
QMI_DMS_GET_DEVICE_HARDWARE_REV	0x002C	Queries the hardware revision of the device.
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.

Table 3-1 QMI\_DMS messages (cont.)

Command	ID	Description
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.
QMI_DMS_UIM_GET_CK_STATUS	0x0040	Queries the status of a UIM facility control key.
QMI_DMS_UIM_SET_CK_PROTECTION	0x0041	Sets the protection of a UIM facility control key.
QMI_DMS_UIM_UNBLOCK_CK	0x0042	Unblocks a UIM facility control key.
QMI_DMS_UIM_GET_IMSI	0x0043	Queries the International Mobile Station Identity (IMSI) of the UIM for the device.
QMI_DMS_UIM_GET_STATE	0x0044	Queries the state of the UIM.
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.



**Table 3-1 QMI\_DMS messages (cont.)**

<b>Command</b>	<b>ID</b>	<b>Description</b>
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.

## 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 occurred 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 last modified
Power State Reporting	1.0
Battery Level Report Limits	1.0
PIN State Reporting	1.1
Activation State Reporting	1.6
Operating Mode Reporting	1.3
UIM State Reporting	1.3
Wireless Disable State Reporting	1.6
PRL Init Reporting	1.7

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	Power State Reporting
Length	1		2	

Field	Field value	Parameter	Size (byte)	Description
<b>Value</b>	→	report_power_state	1	Values: • 0 – Do not report • 1 – Report on change in power state
<b>Type</b>	0x11		1	Battery Level Report Limits
<b>Length</b>	2		2	
<b>Value</b>	→	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).
		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).
<b>Type</b>	0x12		1	PIN State Reporting
<b>Length</b>	1		2	
<b>Value</b>	→	report_pin_state	1	Values: • 0 – Do not report • 1 – Report on change in PIN state
<b>Type</b>	0x13		1	Activation State Reporting
<b>Length</b>	1		2	
<b>Value</b>	→	report_activation_state	1	Values: • 0 – Do not report • 1 – Report activation state changes
<b>Type</b>	0x14		1	Operating Mode Reporting
<b>Length</b>	1		2	
<b>Value</b>	→	report_oprt_mode_state	1	Values: • 0 – Do not report • 1 – Report operating mode changes
<b>Type</b>	0x15		1	UIM State Reporting
<b>Length</b>	1		2	
<b>Value</b>	→	report_uim_state	1	Values: • 0 – Do not report • 1 – Report UIM state changes
<b>Type</b>	0x16		1	Wireless Disable State Reporting
<b>Length</b>	1		2	
<b>Value</b>	→	report_wireless_disable_state	1	Values: • 0 – Do not report • 1 – Report wireless disable state changes
<b>Type</b>	0x17		1	PRL Init Reporting
<b>Length</b>	1		2	
<b>Value</b>	→	report_prl_init	1	Values: • 0 – Do not report • 1 – Report PRL initialized notification

### 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 last modified
Power State	1.0
PIN 1 Status	1.1
PIN 2 Status	1.1
Activation State	1.6

Name	Version last modified
Operating Mode	1.3
UIM State	1.3
Wireless Disable State	1.6
PRL Init Notification	1.7

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x10		1	Power State
<b>Length</b>	2		2	
<b>Value</b>	→	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
		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
<b>Type</b>	0x11		1	PIN 1 Status
<b>Length</b>	3		2	
<b>Value</b>	→	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
		verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.
<b>Type</b>	0x12		1	PIN 2 Status
<b>Length</b>	3		2	

Field	Field value	Parameter	Size (byte)	Description
<b>Value</b>	→	status	1	Current status of the PIN. Values: <ul style="list-style-type: none"> <li>• 0 – PIN is not initialized</li> <li>• 1 – PIN is enabled, not verified</li> <li>• 2 – PIN is enabled, verified</li> <li>• 3 – PIN is disabled</li> <li>• 4 – PIN is blocked</li> <li>• 5 – PIN is permanently blocked</li> <li>• 6 – PIN is unblocked</li> <li>• 7 – PIN is changed</li> </ul>
		verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.
<b>Type</b>	0x13		1	Activation State
<b>Length</b>	2		2	
<b>Value</b>	→	activation_state	2	Service activation state. Values: <ul style="list-style-type: none"> <li>• 0x00 – Service is not activated</li> <li>• 0x01 – Service is activated</li> <li>• 0x02 – Activation connecting – Network connection is in progress for automatic activation of service</li> <li>• 0x03 – Activation connected – Network connection is connected for automatic activation of service</li> <li>• 0x04 – OTASP security is authenticated</li> <li>• 0x05 - OTASP NAM is downloaded</li> <li>• 0x06 - OTASP MDN is downloaded</li> <li>• 0x07 - OTASP IMSI downloaded</li> <li>• 0x08 - OTASP PRL is downloaded</li> <li>• 0x09 - OTASP SPC is downloaded</li> <li>• 0x0A - OTASP settings are committed</li> </ul>
			1	Operating Mode
			2	
<b>Value</b>	→	operating_mode	1	Current operating mode. Values: <ul style="list-style-type: none"> <li>• 0 – Online</li> <li>• 1 – Low power</li> <li>• 2 – Factory test mode</li> <li>• 3 – Offline</li> <li>• 4 – Resetting</li> <li>• 5 – Shutting down</li> <li>• 6 – Persistent low power</li> <li>• 7 – Mode-only low power</li> </ul>
			1	UIM State
			2	



Field	Field value	Parameter	Size (byte)	Description
<b>Value</b>	→	uim_state	1	UIM state. Values: <ul style="list-style-type: none"> <li>• 0x00 – UIM initialization completed</li> <li>• 0x01 – UIM failed</li> <li>• 0x02 – UIM is not present</li> <li>• 0xFF – UIM state is currently unavailable</li> </ul>
<b>Type</b>	0x16		1	Wireless Disable State
<b>Length</b>	1		2	
<b>Value</b>	→	wireless_disable_state	1	Wireless disable state. Values: <ul style="list-style-type: none"> <li>• 0x00 – Wireless disable switch is turned off</li> <li>• 0x01 – Wireless disable switch is turned on</li> </ul>
<b>Type</b>	0x17		1	PRL Init Notification
<b>Length</b>	1		2	
<b>Value</b>	→	prl_init	1	PRL initialized. Values: <ul style="list-style-type: none"> <li>• 0x01 – PRL is completely loaded into the device (could be the default PRL).</li> </ul>

## Error codes

QMI_ERR_NONE	No error occurred 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	Some TLV was 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\_DEVICE\_CAP

Requests the device capabilities.

#### DMS message ID

0x0020

#### Version introduced

Major - 1, Minor - 0

#### 3.3.1 Request - QMI\_DMS\_GET\_DEVICE\_CAP\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.3.2 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 last modified
Device Capabilites	1.4

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Device Capabilites
Length	Var		2	
Value	→	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.
		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.
		data_service_capability	1	Values: <ul style="list-style-type: none"> <li>• 0 – No data services supported</li> <li>• 1 – Only circuit-switched (CS) services are supported</li> <li>• 2 – Only packet-switched (PS) services are supported</li> <li>• 3 – Simultaneous CS and PS</li> <li>• 4 – Nonsimultaneous CS and PS</li> </ul>
		sim_capability	1	Values: <ul style="list-style-type: none"> <li>• 1 – SIM is not supported</li> <li>• 2 – SIM is supported</li> </ul>
		radio_if_list_len	1	Number of sets of the following elements: <ul style="list-style-type: none"> <li>• radio_if_list</li> </ul>
		radio_if_list	Var	List of N one-byte elements describing the radio interfaces supported by the device. Values: <ul style="list-style-type: none"> <li>• 1 – CDMA2000 1X</li> <li>• 2 – CDMA2000 HRPD (1xEV-DO)</li> <li>• 4 – GSM</li> <li>• 5 – UMTS</li> <li>• 8 – LTE</li> </ul>

## Optional TLVs

None

## Error codes

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

### 3.3.3 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 [\[S3\]](#)).

### 3.4 QMI\_DMS\_GET\_DEVICE\_MFR

Requests the device the manufacturer information.

#### DMS message ID

0x0021

#### Version introduced

Major - 1, Minor - 0

#### 3.4.1 Request - QMI\_DMS\_GET\_DEVICE\_MFR\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.4.2 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 last modified
Device Manufacturer	1.0

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

## Optional TLVs

None

## Error codes

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

### 3.4.3 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 [\[S3\]](#)).

## 3.5 QMI\_DMS\_GET\_DEVICE\_MODEL\_ID

Requests the device model identification.

### DMS message ID

0x0022

### Version introduced

Major - 1, Minor - 0

### 3.5.1 Request - QMI\_DMS\_GET\_DEVICE\_MODEL\_ID\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.5.2 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 last modified
Device Model	1.0

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

## Optional TLVs

None

## Error codes

QMI_ERR_NONE	No error in 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\_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 [\[S3\]](#)).

## 3.6 QMI\_DMS\_GET\_DEVICE\_REV\_ID

Requests the device firmware revision identification.

### DMS message ID

0x0023

### Version introduced

Major - 1, Minor - 0

### 3.6.1 Request - QMI\_DMS\_GET\_DEVICE\_REV\_ID\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.6.2 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 last modified
Revision ID	1.0

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

## Optional TLVs

Name	Version last modified
Boot Code Revision	1.6
PRI Revision	1.6

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

## Error codes

QMI_ERR_NONE	No error in equest
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	The device could not allocate memory to formulate a response

### 3.6.3 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

## 3.7 QMI\_DMS\_GET\_MSISDN

Requests the assigned voice number.

### DMS message ID

0x0024

### Version introduced

Major - 1, Minor - 0

### 3.7.1 Request - QMI\_DMS\_GET\_MSISDN\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.7.2 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 last modified
Voice Number	1.0

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

## Optional TLVs

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

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

## Error codes

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

### 3.7.3 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 [\[S3\]](#)).

## 3.8 QMI\_DMS\_GET\_DEVICE\_SERIAL\_NUMBERS

Requests the serial numbers of the device.

### DMS message ID

0x0025

### Version introduced

Major - 1, Minor - 0

### 3.8.1 Request - QMI\_DMS\_GET\_DEVICE\_SERIAL\_NUMBERS\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.8.2 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 last modified
ESN	1.0
IMEI	1.0
MEID	1.0
IMEI SVN	1.5

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

## Error codes

QMI_ERR_NONE	No error in 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

### 3.8.3 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 [\[S3\]](#)).

## 3.9 QMI\_DMS\_GET\_POWER\_STATE

Requests the power status of the device.

### DMS message ID

0x0026

### Version introduced

Major - 1, Minor - 0

### 3.9.1 Request - QMI\_DMS\_GET\_POWER\_STATE\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.9.2 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 last modified
Power State	1.0

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Power State
Length	2		2	
Value	→	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
		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 request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	The device could not allocate memory to formulate a response

### 3.9.3 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 [\[S3\]](#)).

### 3.10 QMI\_DMS\_UIM\_SET\_PIN\_PROTECTION

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

#### DMS message ID

0x0027

#### Version introduced

Major - 1, Minor - 1

#### 3.10.1 Request - QMI\_DMS\_UIM\_SET\_PIN\_PROTECTION\_REQ

#### Message type

Request

#### Sender

Control Point

#### Mandatory TLVs

Name	Version last modified
PIN Protection Information	1.1

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	PIN Protection Information
Length	Var		2	
Value	→	pin_id	1	Specifies the ID of the PIN to be enabled or disabled. Values: <ul style="list-style-type: none"> <li>• 1 – PIN1 (also called PIN)</li> <li>• 2 – PIN2</li> </ul>
		protection_setting_enabled	1	Specifies whether the PIN is enabled. Values: <ul style="list-style-type: none"> <li>• 0 – Disable PIN</li> <li>• 1 – Enable PIN</li> </ul>
		pin_value_len	1	Number of sets of the following elements: <ul style="list-style-type: none"> <li>• pin_value</li> </ul>
		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

**3.10.2 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 last modified
Pin Retries Status	1.1

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x10		1	Pin Retries Status
<b>Length</b>	2		2	
<b>Value</b>	→	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		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 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	Some TLV was 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

### 3.10.3 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.

### 3.11 QMI\_DMS\_UIM\_VERIFY\_PIN

Verifies the PIN before accessing the UIM contents.

#### DMS message ID

0x0028

#### Version introduced

Major - 1, Minor - 1

#### 3.11.1 Request - QMI\_DMS\_UIM\_VERIFY\_PIN\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

Name	Version last modified
PIN Value	1.1

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x01		1	PIN Value
<b>Length</b>	Var		2	
<b>Value</b>	→	pin_id	1	Specifies the ID of the PIN to be enabled or disabled. Values: <ul style="list-style-type: none"> <li>• 1 – PIN1 (also called PIN)</li> <li>• 2 – PIN2</li> </ul>
		pin_value_len	1	Number of sets of the following elements: <ul style="list-style-type: none"> <li>• pin_value</li> </ul>
		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



### 3.11.2 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 last modified
PIN Retries Status	1.1

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	PIN Retries Status
Length	2		2	
Value	→	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		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 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 the 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	Some TLV was 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

### 3.11.3 Description of QMI\_DMS\_UIM\_VERIFY\_PIN REQ/RESP

This command verifies the PIN before accessing the UIM contents.

## 3.12 QMI\_DMS\_UIM\_UNBLOCK\_PIN

Unblocks a blocked PIN.

### DMS message ID

0x0029

### Version introduced

Major - 1, Minor - 1

### 3.12.1 Request - QMI\_DMS\_UIM\_UNBLOCK\_PIN\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
PIN Unblock Information	1.1

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

**Optional TLVs**

None

**3.12.2 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 last modified
PIN Retries Status	1.1

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	PIN Retries Status
Length	2		2	
Value	→	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		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 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	Some TLV was 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\_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.

### 3.13 QMI\_DMS\_UIM\_CHANGE\_PIN

Changes the PIN value.

#### DMS message ID

0x002A

#### Version introduced

Major - 1, Minor - 1

#### 3.13.1 Request - QMI\_DMS\_UIM\_CHANGE\_PIN\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
PIN Change Information	1.1

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

## Optional TLVs

None

### 3.13.2 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 last modified
PIN Retries Status	1.1

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	PIN Retries Status
Length	2		2	
Value	→	verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		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 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	Some TLV was 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\_CHANGE\_PIN REQ/RESP

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



## 3.14 QMI\_DMS\_UIM\_GET\_PIN\_STATUS

Gets the status of a PIN.

### DMS message ID

0x002B

### Version introduced

Major - 1, Minor - 1

### 3.14.1 Request - QMI\_DMS\_UIM\_GET\_PIN\_STATUS\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.14.2 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 last modified
PIN 1 Status	1.1
PIN 2 Status	1.1

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x11		1	PIN 1 Status
<b>Length</b>	3		2	
<b>Value</b>	→	status	1	Current status of the PIN. Values: <ul style="list-style-type: none"> <li>• 0 – PIN is not initialized</li> <li>• 1 – PIN is enabled, not verified</li> <li>• 2 – PIN is enabled, verified</li> <li>• 3 – PIN is disabled</li> <li>• 4 – PIN is blocked</li> <li>• 5 – PIN is permanently blocked</li> <li>• 6 – PIN is unblocked</li> <li>• 7 – PIN is changed</li> </ul>
		verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		unblock_retries_left	1	Number of unblock retries left, after which the PIN is permanently blocked, i.e., the UIM is unusable.
<b>Type</b>	0x12		1	PIN 2 Status
<b>Length</b>	3		2	
<b>Value</b>	→	status	1	Current status of the PIN. Values: <ul style="list-style-type: none"> <li>• 0 – PIN is not initialized</li> <li>• 1 – PIN is enabled, not verified</li> <li>• 2 – PIN is enabled, verified</li> <li>• 3 – PIN is disabled</li> <li>• 4 – PIN is blocked</li> <li>• 5 – PIN is permanently blocked</li> <li>• 6 – PIN is unblocked</li> <li>• 7 – PIN is changed</li> </ul>
		verify_retries_left	1	Number of retries left, after which the PIN is blocked.
		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 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

### 3.14.3 Description of QMI\_DMS\_UIM\_GET\_PIN\_STATUS REQ/RESP

This command returns the status of PIN1 and PIN2.

### 3.15 QMI\_DMS\_GET\_DEVICE\_HARDWARE\_REV

Queries the hardware revision of the device.

#### DMS message ID

0x002C

#### Version introduced

Major - 1, Minor - 2

#### 3.15.1 Request - QMI\_DMS\_GET\_DEVICE\_HARDWARE\_REV\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.15.2 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 last modified
Hardware Revision	1.2

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Hardware Revision
Length	Var		2	
Value	→	hardware_rev	Var	String containing the hardware revision of the device.

## Optional TLVs

None

## Error codes

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

### 3.15.3 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.

## 3.16 QMI\_DMS\_GET\_OPERATING\_MODE

Queries the current operating mode of the device.

### DMS message ID

0x002D

### Version introduced

Major - 1, Minor - 2

### 3.16.1 Request - QMI\_DMS\_GET\_OPERATING\_MODE\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.16.2 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 last modified
Operating Mode	1.2

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x01		1	Operating Mode
<b>Length</b>	1		2	
<b>Value</b>	→	operating_mode	1	Selected operating mode. Values: <ul style="list-style-type: none"> <li>• 0 – Online</li> <li>• 1 – Low power</li> <li>• 2 – Factory Test mode</li> <li>• 3 – Offline</li> <li>• 4 – Resetting</li> <li>• 5 – Shutting down</li> <li>• 6 – Persistent low power</li> </ul>

## Optional TLVs

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

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x10		1	Offline Reason
<b>Length</b>	2		2	
<b>Value</b>	→	offline_reason	2	Offline reason bitmask. All unlisted bits are reserved for future use and are ignored. Values: <ul style="list-style-type: none"> <li>• 0x0001 – Host image misconfiguration</li> <li>• 0x0002 – PRI image misconfiguration</li> <li>• 0x0004 – PRI version incompatible</li> <li>• 0x0008 – Device memory is full, cannot copy PRI information</li> </ul>
<b>Type</b>	0x11		1	Hardware-Restricted Mode
<b>Length</b>	1		2	
<b>Value</b>	→	hardware_controlled_mode	1	Hardware Restricted mode. Values: <ul style="list-style-type: none"> <li>• 0x01 – TRUE</li> </ul>

## Error codes

QMI_ERR_NONE	No error in request
QMI_ERR_INTERNAL	Unexpected error occurred during processing

QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
-------------------	--

### 3.16.3 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

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.



## 3.17 QMI\_DMS\_SET\_OPERATING\_MODE

Sets the operating mode of the device.

### DMS message ID

0x002E

### Version introduced

Major - 1, Minor - 2

### 3.17.1 Request - QMI\_DMS\_SET\_OPERATING\_MODE\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
Operating Mode	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Operating Mode
Length	1		2	
Value	→	operating_mode	1	Selected operating mode. Values: <ul style="list-style-type: none"><li>• 0 – Online</li><li>• 1 – Low power</li><li>• 2 – Factory Test mode</li><li>• 3 – Offline</li><li>• 4 – Resetting</li><li>• 5 – Shutting down</li><li>• 6 – Persistent low power</li><li>• 7 – Mode-only low power</li></ul>

#### Optional TLVs

None

### 3.17.2 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 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	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

### 3.17.3 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.

## 3.18 QMI\_DMS\_GET\_TIME

Queries the current time of the device.

### DMS message ID

0x002F

### Version introduced

Major - 1, Minor - 3

### 3.18.1 Request - QMI\_DMS\_GET\_TIME\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.18.2 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 last modified
Device Time	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Device Time
Length	8		2	
Value	→	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.
		time_source	2	Source of the timestamp. Values: <ul style="list-style-type: none"> <li>• 0 – 32 kHz device clock</li> <li>• 1 – CDMA network</li> <li>• 2 – HDR network</li> </ul>

## Optional TLVs

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

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	System Time in Milliseconds
Length	8		2	
Value	→	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	→	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 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\_TIME REQ/RESP

This message 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.

### 3.19 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

#### 3.19.1 Request - QMI\_DMS\_GET\_PRL\_VER\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.19.2 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 last modified
PRL Version	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	PRL Version
Length	2		2	
Value	→	prl_version	2	PRL version.

### Optional TLVs

Name	Version last modified
PRL-Only Preference	1.5

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	PRL-Only Preference
Length	1		2	
Value	→	prl_only	1	PRL-only preference. Values: <ul style="list-style-type: none"> <li>• 0 – Unset</li> <li>• 1 – Set</li> </ul>

### Error codes

QMI_ERR_NONE	No error in 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

### 3.19.3 Description of QMI\_DMS\_GET\_PRL\_VER REQ/RESP

This message 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 when multiple sessions are available from the UIM elicit a QMI\_ERR\_INFO\_UNAVAILABLE error. The QMI\_UIM service must be used to retrieve the PRL version when this error is returned.

## 3.20 QMI\_DMS\_GET\_ACTIVATION\_STATE

Queries the activation state of the device.

### DMS message ID

0x0031

### Version introduced

Major - 1, Minor - 6

### 3.20.1 Request - QMI\_DMS\_GET\_ACTIVATION\_STATE\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.20.2 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 last modified
Activation State	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Activation State
Length	2		2	
Value	→	activation_state	2	Service activation state. Values: <ul style="list-style-type: none"> <li>• 0x00 – Service is not activated</li> <li>• 0x01 –Service is activated</li> <li>• 0x02 – Activation is connecting - Network connection in progress for automatic activation of service</li> <li>• 0x03 – Activation is connected - Network connection is connected for automatic activation of service</li> <li>• 0x04 – OTASP security is authenticated</li> <li>• 0x05 – OTASP NAM is downloaded</li> <li>• 0x06 – OTASP MDN is downloaded</li> <li>• 0x07 – OTASP IMSI is downloaded</li> <li>• 0x08 – OTASP PRL is downloaded</li> <li>• 0x09 – OTASP SPC is downloaded</li> <li>• 0x0A – OTASP settings are committed</li> </ul>

### Optional TLVs

None

### Error codes

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

### 3.20.3 Description of QMI\_DMS\_GET\_ACTIVATION\_STATE REQ/RESP

This message queries the service activation state of the device.

Note: This is only supported for 3GPP2 devices.

## 3.21 QMI\_DMS\_ACTIVATE\_AUTOMATIC

Requests that the device perform automatic service activation.

### DMS message ID

0x0032

### Version introduced

Major - 1, Minor - 6

### 3.21.1 Request - QMI\_DMS\_ACTIVATE\_AUTOMATIC\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
Activation Code	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Activation Code
Length	Var		2	
Value	→	act_code_len	1	Number of sets of the following elements: <ul style="list-style-type: none"><li>act_code</li></ul>
		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

### 3.21.2 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 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	Some TLV was 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

### 3.21.3 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.

## 3.22 QMI\_DMS\_ACTIVATE\_MANUAL

Requests that the device perform manual service activation.

### DMS message ID

0x0033

### Version introduced

Major - 1, Minor - 6

### 3.22.1 Request - QMI\_DMS\_ACTIVATE\_MANUAL\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
Manual Activation Data	1.6

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



## Optional TLVs

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

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

### 3.22.2 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 request
QMI_ERR_INTERNAL	Unexpected error 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	Some TLV was 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

### 3.22.3 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.

### 3.23 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

#### 3.23.1 Request - QMI\_DMS\_GET\_USER\_LOCK\_STATE\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.23.2 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 last modified
User Lock State	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	User Lock State
Length	1		2	
Value	→	lock_enabled	1	Current state of the lock. Values: <ul style="list-style-type: none"><li>• 0 – Disabled</li><li>• 1 – Enabled</li></ul>

### Optional TLVs

None

### Error codes

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

### 3.23.3 Description of QMI\_DMS\_GET\_USER\_LOCK\_STATE REQ/RESP

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

## 3.24 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

#### 3.24.1 Request - QMI\_DMS\_SET\_USER\_LOCK\_STATE\_REQ

### Message type

Request

### Sender

Control point

### Mandatory TLVs

Name	Version last modified
User Lock State	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	User Lock State
Length	5		2	
Value	→	lock_state	1	Current state of the lock. Values: <ul style="list-style-type: none"><li>• 0 – Disabled</li><li>• 1 – Enabled</li></ul>
		lock_code	4	4-byte code set for the lock in ASCII format (digits 0 to 9 only).

### Optional TLVs

None

### 3.24.2 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 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

### 3.24.3 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.



## 3.25 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

### 3.25.1 Request - QMI\_DMS\_SET\_USER\_LOCK\_CODE\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
User Lock Code	1.3

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

#### Optional TLVs

None

### 3.25.2 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 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

### 3.25.3 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.

## 3.26 QMI\_DMS\_READ\_USER\_DATA

Queries the user data maintained by the device.

### DMS message ID

0x0037

### Version introduced

Major - 1, Minor - 6

### 3.26.1 Request - QMI\_DMS\_READ\_USER\_DATA\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.26.2 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 last modified
User Data	1.6

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

## Optional TLVs

None

## Error codes

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

### 3.26.3 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.

## 3.27 QMI\_DMS\_WRITE\_USER\_DATA

Writes user data maintained by the device.

### DMS message ID

0x0038

### Version introduced

Major - 1, Minor - 6

### 3.27.1 Request - QMI\_DMS\_WRITE\_USER\_DATA\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
User Data	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	User Data
Length	Var		2	
Value	→	data_len	2	Number of sets of the following elements: <ul style="list-style-type: none"><li>• data</li></ul>
		data	Var	User data from/to persistent storage (maximum 512).

#### Optional TLVs

None

### 3.27.2 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 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	Some TLV was 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

### 3.27.3 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 3.26.3).

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.

## 3.28 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

### 3.28.1 Request - QMI\_DMS\_READ\_ERI\_FILE\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.28.2 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 last modified
ERI File	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	ERI File
Length	Var		2	
Value	→	eri_data_len	2	Number of sets of the following elements: • eri_data
		eri_data	Var	ERI data read from persistent storage (maximum 1024).

## Optional TLVs

None

## Error codes

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

### 3.28.3 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.



## 3.29 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

### 3.29.1 Request - QMI\_DMS\_RESTORE\_FACTORY\_DEFAULTS\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
Service Programming Code	1.6

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

#### Optional TLVs

None

### 3.29.2 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 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	Some TLV was 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

**3.29.3 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.

### 3.30 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

#### 3.30.1 Request - QMI\_DMS\_VALIDATE\_SERVICE\_PROGRAMMING\_CODE\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

Name	Version last modified
Programming Code	1.3

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

##### Optional TLVs

None

#### 3.30.2 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 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

**3.30.3 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.

### 3.31 QMI\_DMS\_UIM\_GET\_ICCID

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

#### DMS message ID

0x003C

#### Version introduced

Major - 1, Minor - 3

#### 3.31.1 Request - QMI\_DMS\_UIM\_GET\_ICCID\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.31.2 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 last modified
UIM ICCID	1.3

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

### 3.31.3 Description of QMI\_DMS\_UIM\_GET\_ICCID REQ/RESP

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

## 3.32 QMI\_DMS\_UIM\_GET\_CK\_STATUS

Queries the status of a UIM facility control key.

### DMS message ID

0x0040

### Version introduced

Major - 1, Minor - 3

### 3.32.1 Request - QMI\_DMS\_UIM\_GET\_CK\_STATUS\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
UIM Personalization Facility	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	UIM Personalization Facility
Length	1		2	
Value	→	facility	1	MT or network facility (corresponding AT+CLCK value). Values: <ul style="list-style-type: none"><li>• 0 – Network personalization (PN)</li><li>• 1 – Network subset personalization (PU)</li><li>• 2 – Service provider personalization (PP)</li><li>• 3 – Corporate personalization (PC)</li><li>• 4 – UIM personalization (PF)</li></ul>

#### Optional TLVs

None

### 3.32.2 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 last modified
Facility CK Status	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Facility CK Status
Length	3		2	
Value	→	facility_state	1	UIM facility state. Values: <ul style="list-style-type: none"> <li>• 0 – Deactivated</li> <li>• 1 – Activated</li> <li>• 2 – Blocked</li> </ul>
		verify_retries_left	1	Indicates the number of retries left, after which the CK is blocked.
		unblock_retries_left	1	Number of unblock retries left, after which the CK is permanently blocked.

#### Optional TLVs

Name	Version last modified
Operation Blocking Facility	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	Operation Blocking Facility
Length	1		2	
Value	→	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 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

### 3.32.3 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.

### 3.33 QMI\_DMS\_UIM\_SET\_CK\_PROTECTION

Sets the protection of a UIM facility control key.

#### DMS message ID

0x0041

#### Version introduced

Major - 1, Minor - 3

#### 3.33.1 Request - QMI\_DMS\_UIM\_SET\_CK\_PROTECTION\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
UIM Personalization Facility	1.3

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x01		1	UIM Personalization Facility
<b>Length</b>	Var		2	
<b>Value</b>	→	facility	1	UIM Personalization facility (corresponding AT+CLCK value). Values: <ul style="list-style-type: none"> <li>• 0 – Network personalization (PN)</li> <li>• 1 – Network subset personalization (PU)</li> <li>• 2 – Service provider personalization (PP)</li> <li>• 3 – Corporate personalization (PC)</li> <li>• 4 – UIM personalization (PF)</li> </ul>
		facility_state	1	UIM facility state. Values: <ul style="list-style-type: none"> <li>• 0 – Deactivated</li> </ul>
		facility_ck_len	1	Number of sets of the following elements: <ul style="list-style-type: none"> <li>• facility_ck</li> </ul>
		facility_ck	Var	Facility depersonalization control key string in ASCII text (maximum 8 bytes).

## Optional TLVs

None

### 3.33.2 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 last modified
Facility CK Retry Status	1.3

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

## Error codes

QMI_ERR_NONE	No error in 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

### 3.33.3 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 (see [\[S1\]](#)).

Requests to set the state when the message is not supported by the device elicit a QMI\_ERR\_OP\_DEVICE\_UNsupported error.

### 3.34 QMI\_DMS\_UIM\_UNBLOCK\_CK

Unlocks a UIM facility control key.

#### DMS message ID

0x0042

#### Version introduced

Major - 1, Minor - 3

#### 3.34.1 Request - QMI\_DMS\_UIM\_UNBLOCK\_CK\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
UIM Personalization Facility	1.3

Field	Field value	Parameter	Size (byte)	Description
<b>Type</b>	0x01		1	UIM Personalization Facility
<b>Length</b>	Var		2	
<b>Value</b>	→	facility	1	UIM personalization facility (corresponding AT+CLCK value). Values: <ul style="list-style-type: none"> <li>• 0 – Network personalization (PN)</li> <li>• 1 – Network subset personalization (PU)</li> <li>• 2 – Service provider personalization (PP)</li> <li>• 3 – Corporate personalization (PC)</li> <li>• 4 – UIM personalization (PF)</li> </ul>
		facility_unblock_ck_len	1	Number of sets of the following elements: <ul style="list-style-type: none"> <li>• facility_unblock_ck</li> </ul>
		facility_unblock_ck	Var	Facility control key string in ASCII text (maximum 8 bytes).

## Optional TLVs

None

### 3.34.2 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 last modified
Facility CK Retry Status	1.3

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

## Error codes

QMI_ERR_NONE	No error in 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

### 3.34.3 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.

### 3.35 QMI\_DMS\_UIM\_GET\_IMSI

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

#### DMS message ID

0x0043

#### Version introduced

Major - 1, Minor - 3

#### 3.35.1 Request - QMI\_DMS\_UIM\_GET\_IMSI\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.35.2 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 last modified
International Mobile Subscriber ID	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	International Mobile Subscriber ID
Length	Var		2	
Value	→	imsi	Var	String containing the international mobile subscriber ID.

### Optional TLVs

None

### Error codes

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

### 3.35.3 Description of QMI\_DMS\_UIM\_GET\_IMSI REQ/RESP

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

### 3.36 QMI\_DMS\_UIM\_GET\_STATE

Queries the state of the UIM.

#### DMS message ID

0x0044

#### Version introduced

Major - 1, Minor - 3

#### 3.36.1 Request - QMI\_DMS\_UIM\_GET\_STATE\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.36.2 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 last modified
UIM State	1.3

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	UIM State
Length	1		2	
Value	→	uim_state	1	UIM state. Values: <ul style="list-style-type: none"> <li>• 0x00 – UIM initialization completed</li> <li>• 0x01 – UIM is locked or the UIM failed</li> <li>• 0x02 – UIM is not present</li> <li>• 0x03 – Reserved</li> <li>• 0xFF – UIM state is currently unavailable</li> </ul>

## Optional TLVs

None

## Error codes

QMI_ERR_NONE	No error in 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

### 3.36.3 Description of QMI\_DMS\_UIM\_GET\_STATE REQ/RESP

This message 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.

### 3.37 QMI\_DMS\_GET\_BAND\_CAPABILITY

Queries the band capability of the device.

#### DMS message ID

0x0045

#### Version introduced

Major - 1, Minor - 3

#### 3.37.1 Request - QMI\_DMS\_GET\_BAND\_CAPABILITY\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.37.2 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 last modified
Band Capability	1.3

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

## Optional TLVs

This TLV is present on devices that support LTE bands.

Name	Version last modified
LTE Band Capability	1.8

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	LTE Band Capability
Length	8		2	
Value	→	lte_band_capability	8	Bitmask of LTE bands supported by the device; see Appendix B for the definition of these values.

## Error codes

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

### 3.37.3 Description of QMI\_DMS\_GET\_BAND\_CAPABILITY REQ/RESP

This command queries the band capability of the device.

### 3.38 QMI\_DMS\_GET\_FACTORY\_SKU

Queries the factory provisioned Stock Keeping Unit (SKU).

#### DMS message ID

0x0046

#### Version introduced

Major - 1, Minor - 6

#### 3.38.1 Request - QMI\_DMS\_GET\_FACTORY\_SKU\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.38.2 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 last modified
Factory SKU	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Factory SKU
Length	Var		2	
Value	→	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 request
QMI_ERR_INTERNAL	Unexpected error occurred during processing
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate response
QMI_ERR_OP_DEVICE_UNSUPPORTED	Operation is not supported by the device

### 3.38.3 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.

### 3.39 QMI\_DMS\_SET\_TIME

Sets the time on the device.

#### DMS message ID

0x004B

#### Version introduced

Major - 1, Minor - 4

#### 3.39.1 Request - QMI\_DMS\_SET\_TIME\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

Name	Version last modified
Time	1.4

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Time
Length	8		2	
Value	→	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 last modified
Time Reference Type	1.4

Field	Field value	Parameter	Size (byte)	Description
Type	0x10		1	Time Reference Type

Field	Field value	Parameter	Size (byte)	Description
<b>Length</b>	4		2	
<b>Value</b>	→	time_reference_type	4	Time reference used while setting the time. Values: <ul style="list-style-type: none"> <li>• 0x00000000 – User time</li> <li>• 0x00000001 to 0xFFFFFFFF – Reserved for future extension</li> </ul>

### 3.39.2 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 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

### 3.39.3 Description of QMI\_DMS\_SET\_TIME REQ/RESP

This message 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.

### 3.40 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

#### 3.40.1 Request - QMI\_DMS\_GET\_ALT\_NET\_CONFIG\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

None

##### Optional TLVs

None

#### 3.40.2 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 last modified
Alternative Net Config	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Alternative Net Config
Length	1		2	
Value	→	alt_net_config	1	Alternative network interface configuration. If not provisioned, the Disabled setting is used by the device as default. Values: <ul style="list-style-type: none"> <li>• 0 – Disabled</li> <li>• 1 – Enabled</li> </ul>

## Optional TLVs

None

## Error codes

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

### 3.40.3 Description of QMI\_DMS\_GET\_ALT\_NET\_CONFIG REQ/RESP

This message 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.

## 3.41 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

### 3.41.1 Request - QMI\_DMS\_SET\_ALT\_NET\_CONFIG\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

Name	Version last modified
Alternative Net Configuration	1.6

Field	Field value	Parameter	Size (byte)	Description
Type	0x01		1	Alternative Net Configuration
Length	1		2	
Value	→	alt_net_config	1	Alternative network interface configuration. Values: <ul style="list-style-type: none"><li>• 0 – Disabled</li><li>• 1 – Enabled</li></ul>

#### Optional TLVs

None

### 3.41.2 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 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	Some TLV was 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

### 3.41.3 Description of QMI\_DMS\_SET\_ALT\_NET\_CONFIG REQ/RESP

This message 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.

## 3.42 QMI\_DMS\_GET\_SW\_VERSION

Queries the software version from the device.

### DMS message ID

0x0051

### Version introduced

Major - 1, Minor - 5

### 3.42.1 Request - QMI\_DMS\_GET\_SW\_VERSION\_REQ

#### Message type

Request

#### Sender

Control point

#### Mandatory TLVs

None

#### Optional TLVs

None

### 3.42.2 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 last modified
Software Version Information	1.5

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

## Optional TLVs

None

## Error codes

QMI_ERR_NONE	No error in 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.

### 3.42.3 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.

### 3.43 QMI\_DMS\_SET\_SPC

Changes the service programming code of the device after authentication.

#### DMS message ID

0x0052

#### Version introduced

Major - 1, Minor - 5

#### 3.43.1 Request - QMI\_DMS\_SET\_SPC\_REQ

##### Message type

Request

##### Sender

Control point

##### Mandatory TLVs

Name	Version last modified
Current SPC	1.5
New SPC	1.5

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

##### Optional TLVs

None

### 3.43.2 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 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	Some TLV was 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

### 3.43.3 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.

# 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.)**

<b>Value</b>	<b>Name</b>
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
Bits 21 and 22	Reserved
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
Bits 25 through 31	Reserved
Bit 32	LTE EUTRAN Band 33 UL: 1900-1920; DL: 1900-1920
Bit 33	LTE EUTRAN Band 34 UL: 2010-2025; DL: 2010-2025
Bit 34	LTE EUTRAN Band 35 UL: 1850-1910; DL: 1850-1910
Bit 35	LTE EUTRAN Band 36 UL: 1930-1990; DL: 1930-1990
Bit 36	LTE EUTRAN Band 37 UL: 1910-1930; DL: 1910-1930
Bit 37	LTE EUTRAN Band 38 UL: 2570-2620; DL: 2570-2620
Bit 38	LTE EUTRAN Band 39 UL: 1880-1920; DL: 1880-1920
Bit 39	LTE EUTRAN Band 40 UL: 2300-2400; DL: 2300-2400
Bit 40	LTE EUTRAN Band 41 UL: 2496-2690; DL: 2496-2690

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

<b>Value</b>	<b>Name</b>
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