Qualcomm
TECHNOLOGIES, INC

QMI CAT 2.26 for MPSS.JO.1.0

QMI Card Application Toolkit Spec

80-NV300-11 A

October 13, 2014

Confidential and Proprietary - Qualcomm Technologies, Inc.

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

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

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

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

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

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

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

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

Contents

List of Figures

List of Tables

Revision History

Revision	Date	Description
A	Oct 2014	Initial release. Created from 80-NH952-11 AF. Updates for this revision include minor version 26. Added: • Section 2.4.7 • Figure A-11 Added new message QMI_CAT_GET_CACHED_PROACTIVE_CMD (Section 3.24).

1 Introduction

1.1 Purpose

This specification documents Major Version 2 of the Qualcomm Messaging Interface (QMI) for the Card Application Toolkit (QMI_CAT).

1.2 Scope

This document is intended for software developers who will be using the QMI_CAT. This document provides the following details:

- Theory of operation – Chapter 2 provides the theory of operation for the QMI_CAT. 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_CAT specification.
- Call flows and additional information – Appendix A through Appendix D provide call flows, a list of supplementary Type-Length-Values (TLVs), a table of application responses, and TLVs for envelope commands.

1.3 Conventions

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

Parameter types are indicated by arrows:

→ Designates an input parameter

← Designates an output parameter

↔ Designates a parameter used for both input and output

1.4 References

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

Table 1-1 Reference documents and standards

Ref.	Document	
Qualcomm Technologies		
Q1	*Application Note: Software Glossary for Customers*	CL93-V3077-1
Q2	*Qualcomm MSM Interface (QMI) Architecture*	80-VB816-1
Q3	*QMI UIM for MPSS.JO.1.0, QMI User Identity Module Spec*	80-NV300-12
Standards		
S1	*Smart Cards: Card Application Toolkit (CAT) – Release 4*	ETSI TS 102 223
S2	*Digital Cellular Telecommunications System (Phase 2+) (GSM); Universal Mobile Telecommunications Systems (UMTS); Alphabets and Language-Specific Information*	ETSI TS 123 038
S3	*Language Codes*	ISO 639-2
S4	*Technical Specification Group Core Network and Terminals; Universal Subscriber Identity Module (USIM); Application Toolkit (USAT)*	3GPP TS 31.111
S5	*AT command set for User Equipment (UE)*	3GPP TS 27.007
S6	*Mobile radio interface Layer 3 Specification; Core network protocols; Stage 3*	3GPP TS 24.008
S7	*Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3*	3GPP TS 24.301
S8	*3rd Generation Partnership Project; Technical Specification Group Terminals Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface*	3GPP TS 11.11
S9	*Smart Cards; UICC-Terminal interface; Physical and Logical characteristics*	ETSI TS 102 221
S10	*Smart Cards; UICC-Contactless Front-end (CLF) Interface; Part 1: Physical and data link layer characteristics*	ETSI TS 102 613

1.5 Technical Assistance

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

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

1.6 Acronyms

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

Table 1-2 Acronyms

Acronym	Definition
ACK	acknowledge
ADN	abbreviated dialing number
AP	application processor
AT	access terminal
BC	bearer capability
BCD	binary-coded decimal
BIP	Bearer Independent Protocol
CAT	Card Application Toolkit
CDS	call detail store module
CE	connection element
CLF	contactless front-end
CSD	circuit-switch data
DCS	data coding scheme
DL	download
DTMF	dual-tone multifrequency
EF	elementary file
EPS	evolved packet system
ESTK	enhanced SIM toolkit
EUTRAN	Evolved UMTS Terrestrial Radio Access
EXT	external
GPRS	general packet radio services
GSTK	generic SIM application toolkit
HCI	host controller interface
IMG	image
IMS	IP multimedia subsystem
IP	Internet Protocol
IPv4	IP version 4
IPv6	IP version 6
ISDN	Integrated Services Digital Network
LANG	language
ME	mobile equipment
MT	mobile terminated
NPI	numbering plan identifier
PDN	packet data network
PDP	Packet Data Protocol
PS	packet-switched
QCI	QoS class identifier
QMI	Qualcomm messaging interface
QoS	quality of service
RDI	restricted digital information
RP	Relay layer Protocol

Table 1-2 Acronyms (cont.)

Acronym	Definition
SAT	SIM application toolkit
SCWS	smart card web server
SDU	service data unit
SEL	selector
SIM	subscriber identity module
SMS	short message service
SMS-PP	SMS point-to-point
SS	supplementary services
TCP	Transfer Control Protocol; Transmission Control Protocol
TLV	type-length-value
TON	type of number
TP	terminal profile or Transfer layer Protocol
TPDU	Transfer Protocol data unit
TR	terminal response
UCS2	two-byte universal character set
UDI	unrestricted digital information
UDP	User Datagram Protocol
UE	user equipment
UI	user interface
UICC	universal integrated circuit card
UL	upload
UMTS	universal mobile telecommunications system
URI	universal resource identifier
URL	universal resource locator
USAT	USIM application toolkit
USIM	universal subscriber identity module
USSD	unstructured supplementary services data
UTRAN	UMTS Terrestrial Radio Access

2 Theory of Operation

2.1 Generalized QMI Service Compliance

The QMI_CAT 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 CAT Service Type

CAT is assigned QMI service type 0x0A.

CAT is also assigned QMI service type 0xE0. However, support for type 0xE0 has been deprecated, and its use should be avoided.

2.3 Message Definition Template

2.3.1 Response Message Result TLV

This TLV is present in all Response messages defined in this document. It is not present in the Indication messages.

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

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

2.4 QMI_CAT Fundamental Concepts

2.4.1 Overview of CAT

CAT provides mechanisms that allow applications existing in the Universal Integrated Circuit Card (UICC) to interact and operate with any terminal that supports the specific mechanisms required by the application. For more detail regarding the defined mechanisms and the commands and protocols related to CAT, refer to [S1].

2.4.2 SAT/USAT Proactive Capability

The Subscriber Identity Module (SIM) Application Toolkit (SAT)/Universal Subscriber Identity Module (USIM) Application Toolkit (USAT) proactive capability is a mechanism whereby the UICC can request specific actions to be taken by the Mobile Equipment (ME) by issuing proactive commands, thus establishing and maintaining an interactive dialog with the user and/or communicating with the network or an external device.

The ME informs the UICC of the success of each command issued to it by the UICC, indicates the command details, and, if applicable, adds more specific information. The proactive command set allows the UICC to instruct the ME to:

- Display text supplied by the UICC on the display, with an indication of priority (normal or high) and a defined action (user activity or timeout) to terminate the text display
- Display a text string, obtain the response as a single user keystroke or a string of keys entered by the user, and pass the response to the UICC; if the response is designated as private by the UICC, the ME will not display the user response on the screen
- Set up a voice call to an address with a specific priority, as indicated by the UICC, with all parameters indicated by the UICC
- Set up a data call to an address with specific bearer capability and priority; all parameters are indicated by the UICC
- Send to the network a short message that contains text that is supplied by the UICC to the ME in either packed or unpacked Short Message Service (SMS) 7-bit alphabet or two-byte Universal Character Set (UCS2) alphabet
- Refresh the image (if applicable) of the USIM data contained in the ME memory, entirely or partially, or instruct the ME to completely reinitialize

2.4.3 Proactive Commands Supported via QMI_CAT

The SAT/USAT proactive commands supported via QMI_CAT are:

- Display Text
- Get Inkey
- Get Input
- Play Tone
- Setup Menu
- Select Item
- Send SMS

- Setup Call
- Setup Event List – Events User Activity, Idle Screen Available, Language Selection, Browser Termination, and Host Controller Interface (HCI) Connectivity
- Activate
- Setup Idle Mode Text
- Send DTMF
- Language Notification
- Refresh
- Launch Browser
- Send SS
- Send USSD
- Provide Local Information
- Open Channel
- Close Channel
- Receive Data
- Send Data
- Bearer Independent Protocol Status
- Activate
- End Proactive Session

A control point can register for a notification of each of the supported proactive commands. Once registered for a particular event, the QMI_CAT service passes the proactive command in raw format or decoded format to the control point using the event report mechanism (see Section 3.2). The detailed mechanisms for this processing are as defined in [S1], Section 6.4, and may include a terminal response (see Section 3.6). If there is no control point registered for a given proactive command, the default behavior from QMI_CAT will be to immediately send an error to the card.

There can be only one control point registered for each type of proactive command at the same time. If a control point subsequently tries to register for a proactive command event that has previously been registered, the registration request will fail (see Section 3.2).

Alpha identifier events are for user confirmation and/or display purposes only. Refresh and Setup Event List proactive commands include the event details that are passed to the control point. When a subsequent event indicated in the Setup Event list occurs, this notification is sent via an envelope event download command (see Section 2.4.4). No terminal response is expected for network-related commands Send SMS, Send DTMF, Send SS, Send USSD, Setup Call, Open Channel, Close Channel, Receive Data, Send Data, and End Proactive session proactive commands.

The proactive commands Setup Menu, Setup Idle Mode Text, and Setup Event List are stored by QMI_CAT, regardless of whether a control point has registered for each event. If no registered control point is active, the proactive command is buffered and sent as an event report indication when a control point registers for this event. Note that only the latest proactive command received (for each type) is buffered; older ones are discarded.

The following types of Open Channel proactive commands are supported in the current implementation:

- Open Channel related to packet data service bearer
- Open Channel related to default (network) bearer

2.4.4 Envelope Commands Supported via QMI_CAT

The envelope commands supported via QMI_CAT are:

- Menu Selection
- Event DL – User Activity
- Event DL – Idle Screen Available
- Event DL – Language Selection
- Event DL – Browser Termination
- Event DL – HCI Connectivity
- Event DL – MT Call
- Event DL – Call Connected
- Event DL – Call Disconnected
- Send Call Control
- SMS-PP Data Download

Each of these envelope commands is sent encoded in the raw or decoded format as defined in [S1]. The control point sends these envelope commands as a response to events for which it has previously registered. For all event downloads (listed as Event DL above), the control point must reissue the envelope command request if the envelope response message indicates Card Busy, as defined in [S1], Section 7.

2.4.5 Refresh Command

The refresh command with mode/stage information is only available when QMI_CAT configuration mode is set to 1 (Gobi™ mode). In other QMI_CAT configuration modes, the command with refresh mode/stage information is provided to clients by the QMI_UIM service ([Q3]), and only the Refresh Alpha or Icon TLV is provided to clients by the QMI_CAT service, even if it is a proactive command originated by the card. The reason for this behavior is that the refresh command has a deep impact on the modem, affecting multiple modules. For this reason, the single proactive command is split into various stages and requires logic to combine the responses from many modules.

2.4.6 Routing Events for Third-party IMS Clients on the AP

A control point (e.g., a third-party IP Multimedia Subsystem (IMS) client) can register for full function events with QMI_CAT at runtime using the event report mechanism (see Section 3.2). The same mechanism can be used to disable event routing to a third-party IMS on the Application Processor (AP) and results in restoring the existing event call flow.

When a control point registers for the proactive command for full function event routing, the control point is expected to handle the proactive command completely with no assistance from the modem. A terminal response is expected, instead of user confirmation, for proactive commands registered as full function events.

The proactive commands that support event routings are as follows:

- Setup Call
- Send SMS

- Send SS
- Send USSD
- Send DTMF

Because of the ability of the control point to dynamically register for some of the modem functions, the modem does not propagate the availability of MT Call, Call Connected, and Call Disconnected events in the SETUP_EVENT_LIST command to the control point. For this reason, the client is expected to send the ENVELOPE commands in any case where it is appropriate and required by specifications. If the event for a requested ENVELOPE of the control point is not part of the SETUP_EVENT, the request is ignored and QMI_ERR_INVALID_OPERATION is returned. This is the correct behavior.

2.4.7 Get Cached Proactive Command

Get Cached Proactive Command is only supported when the QMI_CAT configuration mode is either Android or Custom Raw mode.

The supported proactive commands are:

- Setup Menu
- Setup Event List
- Setup Idle Mode Text

Cached proactive commands are available throughout the life cycle of UICC until a UICC reset, hot swap, recovery, or power cycle occurs.

For a call flow diagram of Get Cached Proactive Command, see Figure A-11.

2.5 Configuration File

2.5.1 Mode

Due to limitations in the modem, to handle proactive commands in both decoded and raw formats, the current implementation supports only one method at a time.

The behavior of the QMI_CAT interface is controlled using NV item 65683, which is stored on the device in /nv/item_files/modem/qmi/cat/qmi_cat_mode.

The file has 1 byte only, with the following possible values:

- 0 – QMI_CAT is disabled
- 1 – Indications are in raw format, but only alpha is passed for Send SMS (compatible with Rev 1.0 of the QMI_CAT interface)
- 2 – Indications are in raw format, with complete messages also passed for network-related commands
- 3 – Indications are in decoded format
- 4 – QMI_CAT works in decoded format, but indications are not sent to the control point and must be pulled
- 5 – Indications are in raw format and allow a customizable terminal profile
- 6 – Indications are in decoded format and allow a customizable terminal profile

All other values are reserved for future use.

2.5.2 Customized Terminal Profile

A customizable terminal profile encoded as in [S1], Section 5.2, is used when QMI_CAT configuration mode is set to 5 or 6. The customizable terminal profile is controlled using NV item 65683, which is stored in the file /nv/item_files/modem/qmi/cat/qmi_cat_custom_tp.

2.5.3 VS Service ID

QMI_CAT is assigned QMI service types 0x0A and 0xE0. However, there are cases when it is convenient to disable service type 0xE0, to avoid conflicts with another service using the same service ID. This behavior is controlled using the NV item 66032, stored in the file /nv/item_files/modem/qmi/cat/qmi_cat_vs_id.

The file has 1 byte only, with the following possible values:

- 0 – Disabled
- 1 – Enabled

2.5.4 Null Alpha

Depending on the modem configuration (bit 2 and bit 26 of NV item 65674), QMI_CAT has a different behavior in the case of a network-related command with Null Alpha and no icon:

- When bit 2 = 0 – Indication with a proactive command is sent only when alpha is present and has a length greater than 0
- When bit 2 = 1 and bit 26 = 0 – Indication with a proactive command that contains No Alpha and Null Alpha is sent, and the client is responsible for implementing the correct behavior, as per TS 102 223 ([S1]), and sending the confirmation back to the modem
- When bit 2 = 1 and bit 26 = 1 – Indication with a proactive command that contains No Alpha and Null Alpha is not sent

Note that if the Alpha TLV is present with a length of 0, it means that alpha is a Null Alpha. On the other hand, if there is no Alpha TLV, it means that alpha is missing.

2.5.5 Default Language

When the proactive command Provide Local Information – Language is received and QMI_CAT does not have a client registered to handle it, QMI_CAT automatically sends a response using the value in NV item 69729, which is stored in the file /nv/item_files/modem/qmi/cat/qmi_cat_default_lang. The NV item contains two bytes with the default language coding, which is sent back to the SIM card in the Terminal Response.

If the NV item is not set, QMI_CAT caches the Provide Local Information – Language proactive command and waits for a client to register to handle it.

2.5.6 Setup Call Display Alpha Event Confirmation

How QMI CAT handles the Setup Call Display Alpha event confirmation can be configured using QMI CAT Display Alpha NV item 71588 with the following values:

- 0 – The client processes the Setup Call Display Alpha event and no action is required from QMI CAT.
- 1 – The client does not process the Setup Call Display Alpha event; QMI CAT intercepts the event

and sends an automatic user confirmation with “Display Alpha Confirmation = NO” back to the modem.

- 2 – The client does not process the Setup Call Display Alpha event; QMI CAT intercepts the event and sends an automatic user confirmation with “Display Alpha Confirmation = YES” back to the modem.

All other values are reserved for future use.

If the NV is inactive, the current implementation has a different default behavior based on the value of QMI CAT configuration mode NV item 65683 (see Section 2.5.1):

- QMI CAT mode = 2 – QMI CAT has a default behavior as Display Alpha NV = 2
- QMI CAT mode = 5 – QMI CAT has a default behavior as Display Alpha NV = 1
- All other values of QMI CAT mode – QMI CAT has a default behavior as Display Alpha NV = 0

2.5.7 Blocking SMS-PP Download Envelope Functionality

Because access to the API to send SMS-PP Download Envelope commands to the SIM card can be used for specific attacks, the SMS-PP Download Envelope functionality is blocked by default. Block SMS-PP Envelope NV item 71557 is available for customer who need to enable the SMS-PP Download Envelope functionality. Valid values:

- 1 – SMS-PP Download Envelope functionality is blocked
- 0 – SMS-PP Download Envelope functionality is enabled

Note: This is a security-related NV item and can be updated only when the modem is in Factory mode.

3 QMI_CAT Messages

Table 3-1 QMI_CAT messages

Command	ID	Description
QMI_CAT_RESET	0x0000	Resets the QMI_CAT service state variables of the requesting control point.
QMI_CAT_SET_EVENT_REPORT	0x0001	Sets the QMI_CAT event reporting conditions for the requesting control point.
QMI_CAT_GET_SUPPORTED_MSGS	0x001E	Queries the set of messages implemented by the currently running software.
QMI_CAT_GET_SUPPORTED_FIELDS	0x001F	Queries the fields supported for a single command as implemented by the currently running software.
QMI_CAT_GET_SERVICE_STATE	0x0020	Queries the QMI_CAT service state.
QMI_CAT_SEND_TR	0x0021	Sends the terminal response to the proactive commands coming from the card.
QMI_CAT_SEND_ENVELOPE_CMD	0x0022	Sends an envelope command to the card.
QMI_CAT_GET_EVENT_REPORT	0x0023	Retrieves the last proactive command from the modem.
QMI_CAT_SEND_DECODED_TR	0x0024	Sends the Terminal Response (TR) in decoded format to the proactive commands coming from the card.
QMI_CAT_SEND_DECODED_ENVELOPE_CMD	0x0025	Sends an envelope command in decoded format to the card.
QMI_CAT_EVENT_CONFIRMATION	0x0026	Sends user and icon confirmation for network-related commands.
QMI_CAT_SCWS_OPEN_CHANNEL	0x0027	Sends the Open Channel indication to the Smart Card Web Server (SCWS) agent.
QMI_CAT_SCWS_CLOSE_CHANNEL	0x0028	Sends the Close Channel indication to the SCWS agent.
QMI_CAT_SCWS_SEND_DATA	0x0029	Sends data to the SCWS agent.
QMI_CAT_SCWS_DATA_AVAILABLE	0x002A	Indicates that data is available.
QMI_CAT_SCWS_CHANNEL_STATUS	0x002B	Informs the modem about a change in the channel state.

Table 3-1 QMI_CAT messages (cont.)

Command	ID	Description
QMI_CAT_GET_TERMINAL_PROFILE	0x002C	Retrieves the current modem terminal profile.
QMI_CAT_SET_CONFIGURATION	0x002D	Changes the configuration of the QMI_CAT service.
QMI_CAT_GET_CONFIGURATION	0x002E	Gets the configuration of the QMI_CAT service.
QMI_CAT_GET_CACHED_PROACTIVE_CMD	0x002F	Retrieves a cached proactive command from the modem.

3.1 QMI_CAT_RESET

Resets the QMI_CAT service state variables of the requesting control point.

CAT message ID

0x0000

Version introduced

Major - 1, Minor - 0

3.1.1 Request - QMI_CAT_RESET_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.1.2 Response - QMI_CAT_RESET_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formatted correctly by the control point or the message was corrupted during the transmission

3.1.3 Description of QMI_CAT_RESET REQ/RESP

This message resets the issuing control point's state kept by the service. Each shared state variable may change as a result according to its arbitration policy (see Section 2.4.3). This is equivalent to closing the service and reopening it, although it is done as one operation; hence, the client ID of the requesting control point does not change.

The control point's state variables are changed to their default values before the response is issued.

3.2 QMI_CAT_SET_EVENT_REPORT

Sets the QMI_CAT event reporting conditions for the requesting control point and indicates a QMI_CAT event.

CAT message ID

0x0001

Version introduced

Major - 1, Minor - 0

3.2.1 Request - QMI_CAT_SET_EVENT_REPORT_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

At least one of the following optional TLVs must be included in this request.

Name	Version introduced	Version last modified
Event Reporting Request	1.0	2.23
Decoded Event Reporting Request	2.0	2.23
Slot	2.6	2.20
Full Function Event Reporting Request	2.18	2.19

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Event Reporting Request
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint32	pc_evt_report_req_mask	4	Event report request bitmask: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh/Refresh Alpha (Refresh when QMI_CAT is configured in Gobi mode, Refresh Alpha in other cases) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Provide Local Information – Time • Bit 23 – Clients must set this bit to zero • Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Clients must set this bit to zero Each bit set indicates a request made to QMI_CAT to register the corresponding proactive command to the control point. All unlisted bits are reserved for future use and must be set to zero.
Type	0x11			1	Decoded Event Reporting Request
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint32	pc_dec_evt_report_req_mask	4	Decoded event report request bitmask: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh Alpha (not supported when QMI CAT is configured in Gobi mode) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Clients must set this bit to zero • Bit 23 – Smart Card Web Server • Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Bearer Independent Protocol Status Each bit set indicates a request made to QMI_CAT to register the corresponding proactive command to the control point. All unlisted bits are reserved for future use and must be set to zero.
Type	0x12			1	Slot
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	mask8	slot_mask	1	Slot used for the registration: • Bit 0 – Slot 1 • Bit 1 – Slot 2 • Bit 2 – Slot 3 • Bit 3 – Slot 4 • Bit 4 – Slot 5 All other bits are reserved for future use. If the TLV is missing, the client is implicitly registering for all available slots.
Type	0x13			1	Full Function Event Reporting Request
Length	4			2	
Value	→	mask32	pc_full_func_evt_report_req_mask	4	Full function event report request bitmask: • Bit 0 – Send SMS • Bit 1 – Setup Call • Bit 2 – Send DTMF • Bit 3 – Send SS • Bit 4 – Send USSD Each bit set indicates a request made to QMI_CAT to enable/disable full function capability of the control point for the corresponding proactive command. All unlisted bits are reserved for future use and must be set to zero. The control point must register the corresponding proactive command with a raw or decoded event report bitmask for receiving events.

3.2.2 Response - QMI_CAT_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

The following optional TLVs are present if the result code is QMI_ERR_EVT_REGISTRATION_FAILED.

Name	Version introduced	Version last modified
Proactive Command Event Report Registration Status	1.0	2.23
Proactive Command Decoded Event Report Registration Status	2.0	2.23
Full Function Event Report Registration Status	2.18	2.19

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Proactive Command Event Report Registration Status
Length	4			2	
Value	→	uint32	pc_evt_report_req_err_mask	4	Proactive command event report registration error bitmask: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh/Refresh Alpha (Refresh when QMI_CAT is configured in Gobi mode, Refresh Alpha in other cases) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Provide Local Information – Time

Field	Field value	Field type	Parameter	Size (byte)	Description
			pc_evt_report_req_err_ mask (cont.)		• Bit 23 – Clients are to ignore this bit • Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Clients are to ignore this bit A set bit indicates that the corresponding proactive command has already been registered by another control point. If a bit that was not set by the control point is included, the control point is to ignore the bit.
Type	0x11			1	Proactive Command Decoded Event Report Registration Status
Length	4			2	
Value	→	uint32	pc_dec_evt_report_req_ err_mask	4	Proactive command decoded event report registration error bitmask: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh Alpha (not supported when QMI CAT is configured in Gobi mode) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Clients are to ignore this bit • Bit 23 – Smart Card Web Server

Field	Field value	Field type	Parameter	Size (byte)	Description
			pc_dec_evt_report_req_ err_mask (cont.)		• Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Bearer Independent Protocol Status A set bit indicates that the corresponding proactive command has already been registered by another control point. If a bit that was not set by the control point is included, the control point is to ignore the bit.
Type	0x12			1	Full Function Event Report Registration Status
Length	4			2	
Value	→	mask32	pc_full_func_evt_report_ err_mask	4	Full function event report request bitmask: • Bit 0 – Send SMS • Bit 1 – Setup Call • Bit 2 – Send DTMF • Bit 3 – Send SS • Bit 4 – Send USSD A set bit indicates that QMI_CAT failed to enable/disable full function capability handling for the corresponding proactive command. If a bit that was not set by the control point is included, the control point is to ignore the bit.

3.2.3 Indication - QMI_CAT_EVENT_REPORT_IND

Message type

Indication

Sender

Service

Scope

Unicast (per control point)

Mandatory TLVs

None

Optional TLVs

One or more of the following optional TLVs must be included in this indication.

Name	Version introduced	Version last modified
Display Text Event	1.0	1.0
Get Inkey Event	1.0	1.0
Get Input Event	1.0	1.0
Setup Menu Event	1.0	1.0
Select Item Event	1.0	1.0
Alpha Identifier Available	1.0	1.0
Setup Event List Event	1.0	1.0
Setup Idle Mode Text Event	1.0	1.0
Language Notification Event	1.0	1.0
Refresh Event	1.0	1.0
End Proactive Session	1.0	1.0
Decoded Header ID	2.0	2.23
Text String	2.0	2.0
High Priority	2.0	2.0
User Control	2.0	2.0
Icon	2.0	2.0
Duration	2.0	2.0
Response Format	2.0	2.0
Help Available	2.0	2.0
Response Packing Format	2.0	2.0
Response Length	2.0	2.0
Show User Input	2.0	2.0
Tone	2.0	2.9
Softkey Selection	2.0	2.0
Items	2.0	2.0
Default Item	2.0	2.0
Next Action Indicator	2.0	2.0
Icon ID List	2.0	2.12
Presentation	2.0	2.0
Packing Required	2.0	2.0
SMS TPDU	2.0	2.0
Is CDMA SMS	2.0	2.0
Address	2.0	2.0
Call Setup Requirement	2.0	2.0
Redial	2.0	2.0
Subaddress	2.0	2.0
Capability Configuration	2.0	2.0
DTMF	2.0	2.0
Specific Language Notification	2.0	2.0
Language	2.0	2.0

Name	Version introduced	Version last modified
Launch Mode	2.0	2.0
URL	2.0	2.0
Browser ID	2.0	2.0
Bearer List	2.0	2.0
Provisioning Files	2.0	2.0
USSD String	2.0	2.0
Default Text	2.0	2.0
Immediate Response Required	2.0	2.0
User Confirmation Alpha	2.0	2.0
Setup Call Display Alpha	2.0	2.0
User Confirmation Icon	2.0	2.0
Setup Call Display Icon	2.0	2.0
Gateway Proxy	2.0	2.0
Alpha	2.0	2.0
Notification Required	2.0	2.0
Play Tone Event	2.2	2.2
Setup Call Event	2.2	2.2
Send DTMF Event	2.2	2.2
Launch Browser Event	2.2	2.2
Send SMS Event	2.2	2.2
Send SS Event	2.2	2.2
Send USSD Event	2.2	2.2
Provide Local Information Event	2.2	2.2
Setup Event List Raw Event	2.2	2.2
Slot	2.2	2.20
Open Channel Event	2.3	2.3
Close Channel Event	2.3	2.3
Send Data Event	2.3	2.3
Receive Data Event	2.3	2.3
On Demand Link Establish	2.4	2.4
CSD Bearer Description	2.4	2.4
GPRS Bearer Description	2.4	2.4
EUTRAN External Parameter Bearer Description	2.4	2.4
EUTRAN External Mapped UTRAN PS Bearer Description	2.4	2.4
Buffer Size	2.4	2.4
Network Access Name	2.4	2.4
Other Address	2.4	2.4
User Login	2.4	2.4
User Password	2.4	2.4
Transport Level	2.4	2.4
Data Destination Address	2.4	2.4
Channel Data Length	2.4	2.4
Send Data Immediately	2.4	2.4
Channel Data	2.4	2.4
Channel ID	2.4	2.4
Items with DCS	2.8	2.8

Name	Version introduced	Version last modified
Activate Event	2.9	2.9
Activate Descriptor Target	2.9	2.9
Response Type	2.18	2.18
Bearer Independent Protocol Status	2.22	2.22
Refresh Alpha	2.23	2.23

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Display Text Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_display_text
		opaque	pc_display_text	Var	Display Text proactive command, encoded as in [S1], Section 6.6.1.
Type	0x11			1	Get Inkey Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_get_inkey
		opaque	pc_get_inkey	Var	Get Inkey proactive command, encoded as in [S1], Section 6.6.2.
Type	0x12			1	Get Input Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_get_input
		opaque	pc_get_input	Var	Get Input proactive command, encoded as in [S1], Section 6.6.3.
Type	0x13			1	Setup Menu Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_menu
		opaque	pc_setup_menu	Var	Setup Menu proactive command, encoded as in [S1], Section 6.6.7.
Type	0x14			1	Select Item Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_select_item
		opaque	pc_select_item	Var	Select Item proactive command, encoded as in [S1], Section 6.6.8.

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x15			1	Alpha Identifier Available (used only when QMI_CAT is configured in Gobi mode)
Length	Var			2	
Value	→	uint8	pc_cmd_type	1	Proactive command type that includes the alpha identifier: • 0x01 – Sends an SMS proactive command All other values are reserved.
		uint16	alpha_id_len	2	Number of sets of the following elements: • alpha_identifier
		opaque	alpha_identifier	Var	Alpha identifier, as in [S1], Section 8.2.
Type	0x16			1	Setup Event List Event (used only when QMI_CAT is configured in Gobi mode)
Length	4			2	
Value	→	uint32	pc_setup_evt_list	4	Setup event list bitmask: • Bit 0 – User Activity Notify • Bit 1 – Idle Screen Available • Bit 2 – Language Selection Notify Each set bit indicates the availability of the corresponding event in the Setup Event list proactive command. All unlisted bits are reserved for future use and are ignored.
Type	0x17			1	Setup Idle Mode Text Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_idle_mode_text
		opaque	pc_setup_idle_mode_text	Var	Setup Idle mode text proactive command, encoded as in [S1], Section 6.6.22.
Type	0x18			1	Language Notification Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_lang_notification
		opaque	pc_lang_notification	Var	Language Notification proactive command, encoded as in [S1], Section 6.6.25.
Type	0x19			1	Refresh Event (used only when QMI_CAT is configured in Gobi mode)

Field	Field value	Field type	Parameter	Size (byte)	Description
Length	4			2	
Value	→	uint16	refresh_mode	2	As indicated in [S1], Section 8.6.
		enum16	refresh_stage	2	Stage of the refresh procedure: • 0x01 – Refresh start • 0x02 – Refresh success • 0x03 – Refresh failed
Type	0x1A			1	End Proactive Session
Length	1			2	
Value	→	enum8	proactive_session_end_type	1	Proactive session end type: • 0x01 – End proactive session command received from the card • 0x02 – End proactive session internal to the ME
Type	0x1B			1	Decoded Header ID
Length	6			2	
Value	→	enum8	command_id	1	ID of the proactive command: • 0x01 – Display Text • 0x02 – Get Inkey • 0x03 – Get Input • 0x04 – Launch Browser • 0x05 – Play Tone • 0x06 – Select Item • 0x07 – Send SMS • 0x08 – Send SS • 0x09 – Send USSD • 0x0A – Setup Call – User Confirmation • 0x0B – Setup Call – Alpha Display • 0x0C – Setup Menu • 0x0D – Setup Idle Text • 0x0E – Provide Local Information – Language • 0x0F – Send DTMF • 0x10 – Language Notification • 0x11 – Setup Event – User Activity • 0x12 – Setup Event – Idle Screen Notify • 0x13 – Setup Event – Language Selection Notify • 0x14 – Open Channel • 0x15 – Close Channel • 0x16 – Receive Data • 0x17 – Send Data • 0x18 – Activate • 0x19 – Setup Event – HCI Connectivity • 0x1A – Refresh Alpha • 0X20 – Setup Event – Browser Termination All other values are reserved.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint32	uim_ref_id	4	Proactive command reference ID (used internally by the QMI_CAT service).
		uint8	command_number	1	Command number sent to the client in the proactive command for tracking purposes to match with the command number in the terminal response.
Type	0x1C			1	Text String
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x1D			1	High Priority
Length	1			2	
Value	→	enum8	high_priority	1	High priority value: • 0x00 – Do not clear the screen • 0x01 – Clear anything that is on the screen
Type	0x1E			1	User Control
Length	1			2	
Value	→	enum8	user_control	1	User control: • 0x00 – Do not allow the user to clear the screen • 0x01 – Allow the user to clear the screen
Type	0x1F			1	Icon
Length	Var			2	
Value	→	enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	rec_num	1	Record number in the EF-IMG file.
		uint16	data_size	2	Number of sets of the following elements: • data
		opaque	data	Var	Image instance data in binary format.
Type	0x20			1	Duration
Length	2			2	
Value	→	enum8	units	1	Time units: • 0x00 – Minutes • 0x01 – Seconds • 0x02 – Tenths of seconds
		uint8	interval	1	Time interval; this number must be greater than zero (see [S1], Section 8.8).
Type	0x21			1	Response Format
Length	1			2	
Value	→	enum8	response_format	1	Response format: • 0x00 – SMS default alphabet • 0x01 – Yes/No • 0x02 – Numerical only • 0x03 – UCS2 • 0x04 – Immediate digit response • 0x05 – Yes/No and immediate digit response
Type	0x22			1	Help Available
Length	1			2	
Value	→	boolean	help_available	1	Whether help is available: • 0x00 – No help is available • 0x01 – Help is available
Type	0x23			1	Response Packing Format
Length	1			2	
Value	→	enum8	response_packing_format	1	Response packing format: • 0x00 – Unpacked format • 0x01 – Packed format
Type	0x24			1	Response Length
Length	2			2	
Value	→	uint8	maximum_user_input	1	Maximum user input. A value of 0xFF indicates no maximum.
		uint8	minimum_user_input	1	Minimum user input. A value of 0x00 indicates no minimum.
Type	0x25			1	Show User Input
Length	1			2	
Value	→	enum8	show_user_input	1	Show user input: • 0x00 – ME can show * characters • 0x01 – ME can show user input
Type	0x26			1	Tone
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	tone	1	Tone to be played: • 0x01 – Dial tone • 0x02 – Called subscriber busy tone • 0x03 – Congestion tone • 0x04 – Radio path ACK tone • 0x05 – Radio path not available, call drop tone • 0x06 – Error tone • 0x07 – Call waiting tone • 0x08 – Ringing tone • 0x09 – General beep • 0x0A – Positive ACK tone • 0x0B – Negative ACK tone • 0x0C – Ring tone selected by the user • 0x0D – SMS alert tone selected by the user • -1 – Not in use
Type	0x27			1	Softkey Selection
Length	1			2	
Value	→	enum8	softkey_selection	1	Softkey selection: • 0x00 – Softkey is not selected • 0x01 – Softkey is selected
Type	0x28			1	Items
Length	Var			2	
Value	→	uint8	number_of_items	1	Number of sets of the following elements: • item_id • item_text_length • item_text
		uint8	item_id	1	ID of the item. Each item has a unique identifier from 0x01 to 0xFF.
		uint8	item_text_length	1	Number of sets of the following elements: • item_text
		opaque	item_text	Var	Item text. Coded the same way that alpha is coded in the EF-ADN file (see [S6], clause 4.4.2.3).
Type	0x29			1	Default Item
Length	1			2	
Value	→	uint8	default_item	1	Default item to be selected. All values are valid, except 0xFF, which is reserved (see [S1], Section 8.10).
Type	0x2A			1	Next Action Indicator
Length	Var			2	
Value	→	uint8	num_of_items	1	Number of sets of the following elements: • next_action_list

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum8	next_action_list	Var	Item in the action list: • 0x00 – Setup Call • 0x01 – Send SS • 0x02 – Send USSD • 0x03 – Send Short Message • 0x04 – Launch Browser • 0x05 – Play Tone • 0x06 – Display Text • 0x07 – Get Inkey • 0x08 – Get Input • 0x09 – Select Item • 0x0A – Setup Menu • 0x0B – Setup Idle Mode Text • 0x0C – End of the Proactive Session • 0x0D – Provide Local Information
Type	0x2B			1	Icon ID List
Length	Var			2	
Value	→	boolean	display_icon_only	1	Whether to display the icon only: • 0x00 – Icon is not self-explanatory, display icon with description • 0x01 – Icon is self-explanatory, display only the icon
		uint8	num_of_items	1	Number of sets of the following elements: • qualifier • height • width • ics • rec_num • data_size • data
		enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color
		uint8	rec_num	1	Record number in the EF-IMG file.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint16	data_size	2	Number of sets of the following elements: • data
		opaque	data	Var	Image instance data in binary format.
Type	0x2C			1	Presentation
Length	1			2	
Value	→	enum8	presentation	1	Presentation type: • 0x00 – Not specified • 0x01 – Data value presentation • 0x02 – Navigation presentation
Type	0x2D			1	Packing Required
Length	1			2	
Value	→	boolean	packing_required	1	Indicates whether packing is required: • 0x00 – Packing is not required • 0x01 – Packing is required
Type	0x2E			1	SMS TPDU
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • sms_tpdu
		opaque	sms_tpdu	Var	SMS TPDU data, as specified in [S6].
Type	0x2F			1	Is CDMA SMS
Length	1			2	
Value	→	boolean	is_cdma_sms	1	CDMA SMS format indication: • 0x00 – FALSE (3GPP format) • 0x01 – TRUE (3GPP2 format) This defaults to FALSE if the TLV is not present.
Type	0x30			1	Address
Length	Var			2	
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved
		uint8	length	1	Number of sets of the following elements: • address_data

Field	Field value	Field type	Parameter	Size (byte)	Description
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x31			1	Call Setup Requirement
Length	1			2	
Value	→	enum8	call_setup_requirement	1	Call setup requirements: • 0x00 – No other calls • 0x01 – Hold active calls • 0x02 – Disconnect active calls
Type	0x32			1	Redial
Length	3			2	
Value	→	boolean	redial_necessary	1	Indicates whether redial is necessary: • 0x00 – Redial is not necessary • 0x01 – Redial is necessary
		enum8	units	1	Time units: • 0x00 – Minutes • 0x01 – Seconds • 0x02 – Tenths of seconds
		uint8	interval	1	Time interval. This value must be greater than zero if redial_necessary is set to 0x01 (see [S1], Section 8.8).
Type	0x33			1	Subaddress
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • subaddress
		opaque	subaddress	Var	Subaddress in BCD format (two digits encoded in one byte). Maximum size of the subaddress is 20 bytes (see [S1], Section 8.3).
Type	0x34			1	Capability Configuration
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • capability_config_data
		opaque	capability_config_data	Var	Capability configuration data (see [S1], Section 8.4).
Type	0x35			1	DTMF
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • dtmf_data
		opaque	dtmf_data	Var	DTMF data in BCD format (two digits encoded in one byte) (see [S1], Section 8.44).
Type	0x36			1	Specific Language Notification
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	boolean	spec_lang_notify	1	Whether there is a specific language notification: • 0x00 – No • 0x01 – Yes
Type	0x37			1	Language
Length	2			2	
Value	→	uint16	language	2	Language value. Each language code is a pair of alphanumeric characters (defined in [S3]). Each alphanumeric character is coded on one byte using the SMS default 7-bit coded alphabet, as defined in [S1], Section 8.45, with bit 8 set to 0.
Type	0x38			1	Launch Mode
Length	1			2	
Value	→	enum8	launch_mode	1	Launch mode: • 0x00 – Launch if not already launched • 0x01 – Use the existing browser • 0x02 – Close the existing browser
Type	0x39			1	URL
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • url_data
		opaque	url_data	Var	URL (see [S1], Section 8.48).
Type	0x3A			1	Browser ID
Length	1			2	
Value	→	uint8	browser_id	1	Browser ID (see [S1], Section 8.47).
Type	0x3B			1	Bearer List
Length	Var			2	
Value	→	uint16	length	2	Number of sets of the following elements: • bearer_list
		enum8	bearer_list	Var	Bearer list: • 0x00 – SMS • 0x01 – CSD • 0x02 – USSD bearer code • 0x03 – GPRS • 0x04 – Bearer default
Type	0x3C			1	Provisioning Files
Length	Var			2	
Value	→	uint32	num_of_prov_files	4	Number of sets of the following elements: • length • path
		uint8	length	1	Number of sets of the following elements: • path

Field	Field value	Field type	Parameter	Size (byte)	Description
		opaque	path	Var	Path to the provisioning file (see [S1], Section 8.50).
Type	0x3D			1	USSD String
Length	Var			2	
Value	→	enum8	orig_dcs_from_sim	1	Original data coding scheme from the SIM: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length	1	Number of sets of the following elements: • text
		opaque	text	Var	Text of USSD string (see [S4], Section 8.17).
Type	0x3E			1	Default Text
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x3F			1	Immediate Response Required
Length	1			2	
Value	→	boolean	immediate_resp	1	Indicates whether an immediate response is required: • 0x00 – No • 0x01 – Yes
Type	0x40			1	User Confirmation Alpha
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x41			1	Setup Call Display Alpha

Field	Field value	Field type	Parameter	Size (byte)	Description
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x42			1	User Confirmation Icon
Length	Var			2	
Value	→	enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color
		uint8	rec_num	1	Record number in the EF-IMG file.
		uint16	data_size	2	Number of sets of the following elements: • data
		opaque	data	Var	Image instance data in binary format.
Type	0x43			1	Setup Call Display Icon
Length	Var			2	
Value	→	enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color
		uint8	rec_num	1	Record number in the EF-IMG file.
		uint16	data_size	2	Number of sets of the following elements: • data
		opaque	data	Var	Image instance data in binary format.
Type	0x44			1	Gateway Proxy
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x45			1	Alpha
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x46			1	Notification Required
Length	1			2	
Value	→	boolean	notification_required	1	Indicates whether the notification for a setup event list is required: • 0 – Notification is not required • 1 – Notification is required
Type	0x47			1	Play Tone Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_play_tone
		opaque	pc_play_tone	Var	Play Tone proactive command, encoded as in [S1], Section 6.6.5.
Type	0x48			1	Setup Call Event
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_call
		opaque	pc_setup_call	Var	Setup Call proactive command, encoded as in [S1], Section 6.6.12.
Type	0x49			1	Send DTMF Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_dtmf
		opaque	pc_send_dtmf	Var	Send DTMF proactive command, encoded as in [S1], Section 6.6.24.
Type	0x4A			1	Launch Browser Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_launch_browser
		opaque	pc_launch_browser	Var	Launch Browser proactive command, encoded as in [S1], Section 6.6.26.
Type	0x4B			1	Send SMS Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_sms
		opaque	pc_send_sms	Var	Send SMS proactive command, encoded as in [S1], Section 6.6.9.
Type	0x4C			1	Send SS Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_ss
		opaque	pc_send_ss	Var	Send SS proactive command, encoded as in [S1], Section 6.6.10.
Type	0x4D			1	Send USSD Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_ussd
		opaque	pc_send_ussd	Var	Send USSD proactive command, encoded as in [S1], Section 6.6.11.
Type	0x4E			1	Provide Local Information Event

Field	Field value	Field type	Parameter	Size (byte)	Description
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_provide_local_info
		opaque	pc_provide_local_info	Var	Provide Local Information proactive command, encoded as in [S1], Section 6.6.15.
Type	0x4F			1	Setup Event List Raw Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_event_list
		opaque	pc_setup_event_list	Var	Setup Event List proactive command, encoded as in [S1], Section 6.6.16.
Type	0x50			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.
Type	0x51			1	Open Channel Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_open_channel
		opaque	pc_open_channel	Var	Open Channel proactive command, encoded as in [S1], Section 6.6.27.
Type	0x52			1	Close Channel Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_close_channel
		opaque	pc_close_channel	Var	Close Channel proactive command, encoded as in [S1], Section 6.6.28.
Type	0x53			1	Send Data Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_data

Field	Field value	Field type	Parameter	Size (byte)	Description
		opaque	pc_send_data	Var	Send Data proactive command, encoded as in [S1], Section 6.6.30.
Type	0x54			1	Receive Data Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_receive_data
		opaque	pc_receive_data	Var	Receive Data proactive command, encoded as in [S1], Section 6.6.29.
Type	0x55			1	On Demand Link Establish
Length	1			2	
Value	→	boolean	on_demand_link_est	1	Indicates whether the link is required: • 0x00 – Link is not required • 0x01 – Link is required
Type	0x56			1	CSD Bearer Description
Length	3			2	
Value	→	uint8	speed	1	Data rate; same as the speed subparameter defined in [S5], Section 6.7.
		enum8	name	1	CSD bearer name: • 0x00 – Data Circuit Asynchronous; UDI or 3.1 kHz modem • 0x01 – Data Circuit Synchronous; UDI or 3.1 kHz modem • 0x02 – PAD Access Asynchronous UDI • 0x03 – Packet Access Synchronous UDI • 0x04 – Data Circuit Asynchronous RDI • 0x05 – Data Circuit Synchronous RDI • 0x06 – PAD Access Asynchronous RDI • 0x07 – Packet Access Synchronous RDI
		enum8	connection_element	1	CSD bearer connection element: • 0x00 – Transparent • 0x01 – Nontransparent • 0x02 – Both, transparent preferred • 0x03 – Both, nontransparent preferred
Type	0x57			1	GPRS Bearer Description
Length	6			2	
Value	→	uint8	precedence_cls	1	Precedence class; same as the precedence subparameter defined in [S4], Section 8.52.2.
		uint8	delay_cls	1	Delay class; same as the delay subparameter defined in [S4], Section 8.52.2.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	reliability_cls	1	Reliability class; same as the reliability subparameter defined in [S4], Section 8.52.2.
		uint8	peak_throughput	1	Peak throughput class; same as the peak subparameter defined in [S4], Section 8.52.2.
		uint8	mean_throughput	1	Mean throughput class; same as the mean subparameter defined in [S4], Section 8.52.2.
		enum8	pkt_data_protocol	1	Packet Data Protocol: • 0x02 – IP All other values are reserved.
Type	0x58			1	EUTRAN External Parameter Bearer Description
Length	17			2	
Value	→	enum8	traffic_class	1	Indicates the type of application for which the UMTS bearer service is optimized: • 0x00 – Conversational • 0x01 – Streaming • 0x02 – Interactive • 0x03 – Background • 0x04 – Subscribed value All other values are reserved.
		uint16	max_bitrate_ul	2	Maximum bitrate UL; same as the maximum bitrate UL subparameter defined in [S4], Section 8.52.3.
		uint16	max_bitrate_dl	2	Maximum bitrate DL; same as the maximum bitrate DL subparameter defined in [S4], Section 8.52.3.
		uint16	guaranteed_bitrate_ul	2	Guaranteed bitrate UL; same as the guaranteed bitrate UL subparameter defined in [S4], Section 8.52.3.
		uint16	guaranteed_bitrate_dl	2	Guaranteed bitrate DL; same as the guaranteed bitrate DL subparameter defined in [S4], Section 8.52.3.
		enum8	delivery_order	1	Numeric parameter that indicates if the UMTS bearer will provide in-sequence SDU delivery: • 0x00 – No • 0x01 – Yes • 0x02 – Subscribed value All other values are reserved.
		uint8	max_sdu_size	1	Maximum SDU size; same as the Maximum SDU size subparameter defined in [S4], Section 8.52.3.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	max_sdu_err_ratio	1	SDU error ratio; same as the SDU error ratio subparameter defined in [S4], Section 8.52.3.
		uint8	residual_bit_err_ratio	1	Residual bit error ratio; same as the residual bit error ratio subparameter defined in [S4], Section 8.52.3.
		enum8	delivery_of_err_sdu	1	Numeric parameter that indicates if SDUs detected as erroneous will be delivered: • 0x00 – No • 0x01 – Yes • 0x02 – No detect • 0x03 – Subscribed value All other values are reserved.
		uint8	transfer_delay	1	Transfer delay; same as the transfer delay subparameter defined in [S4], Section 8.52.3.
		uint8	traffic_handling_pri	1	Traffic handling priority; same as the traffic handling priority subparameter defined in [S4], Section 8.52.3.
		enum8	pdp_type	1	PDP type: • 0x02 – IP All other values are reserved.
Type	0x59			1	EUTRAN External Mapped UTRAN PS Bearer Description
Length	10			2	
Value	→	uint8	qci	1	QCI (see [S4], Section 8.52.5).
		uint8	max_bitrate_ul	1	Maximum bitrate UL (see [S4], Section 8.52.5).
		uint8	max_bitrate_dl	1	Maximum bitrate DL (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_ul	1	Guaranteed bitrate UL (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_dl	1	Guaranteed bitrate DL (see [S4], Section 8.52.5).
		uint8	max_bitrate_ul_ext	1	Maximum bitrate UL Ext (see [S4], Section 8.52.5).
		uint8	max_bitrate_dl_ext	1	Maximum bitrate DL Ext (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_ul_ext	1	Guaranteed bitrate UL Ext (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_dl_ext	1	Guaranteed bitrate DL Ext (see [S4], Section 8.52.5).
		enum8	pdp_type	1	PDP type: • 0x02 – IP All other values are reserved.
Type	0x5A			1	Buffer Size

Field	Field value	Field type	Parameter	Size (byte)	Description
Length	2			2	
Value	→	uint16	buffer_size	2	Buffer size.
Type	0x5B			1	Network Access Name
Length	Var			2	
Value	→	uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Network access name encoded in ASCII character (see [S4], Section 8.61).
Type	0x5C			1	Other Address
Length	Var			2	
Value	→	enum8	address_type	1	Address type: • 0x01 – No address given • 0x02 – Dynamic • 0x03 – IPv4 • 0x04 – IPv6 All other values are reserved.
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address (see [S1], Section 8.58).
Type	0x5D			1	User Login
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x5E			1	User Password
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x5F			1	Transport Level
Length	3			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	transport_protocol	1	Transport protocol: • 0x00 – Not present • 0x01 – UDP • 0x02 – TCP All other values are reserved.
		uint16	port_number	2	Port number.
Type	0x60			1	Data Destination Address
Length	Var			2	
Value	→	enum8	address_type	1	Address type: • 0x01 – No address given • 0x02 – Dynamic • 0x03 – IPv4 • 0x04 – IPv6 All other values are reserved.
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address (see [S1], Section 8.58).
Type	0x61			1	Channel Data Length
Length	1			2	
Value	→	uint8	ch_data_length	1	Number of bytes that are available in the channel buffer, or the number of bytes that are requested in a Received Data command (see [S1], Section 8.54).
Type	0x62			1	Send Data Immediately
Length	1			2	
Value	→	boolean	send_data_immediately	1	Indicates whether to send the data immediately: • 0x00 – No, store the data in the Tx buffer • 0x01 – Yes, send the data immediately
Type	0x63			1	Channel Data
Length	Var			2	
Value	→	uint16	data_len	2	Number of sets of the following elements: • channel_data_string
		opaque	channel_data_string	Var	Channel data string is considered by the terminal as binary coded on 8 bits (see [S1], Section 8.53).
Type	0x64			1	Channel ID
Length	1			2	
Value	→	uint8	ch_id	1	Channel ID (see [S1], Section 8.7).
Type	0x65			1	Items with DCS
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint8	number_of_items	1	Number of sets of the following elements: • item_id • dcs • item_text_length • item_text
		uint8	item_id	1	ID of the item. Each item has a unique identifier from 0x01 to 0xFF.
		enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	item_text_length	1	Number of sets of the following elements: • item_text
		opaque	item_text	Var	Item text (see [S6], clause 4.4.2.3).
Type	0x66			1	Activate Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	pc_activate_len	2	Number of sets of the following elements: • pc_activate
		opaque	pc_activate	Var	Activate proactive command encoded as in [S1], Section 6.6.40.
Type	0x67			1	Activate Descriptor Target
Length	1			2	
Value	→	enum8	target	1	Activate descriptor target (see [S1], Section 8.89): • 0x01 – UICC-CLF interface according to [S10] All other values are reserved for future use.
Type	0x68			1	Response Type
Length	4			2	
Value	→	enum	rsp_type	4	Response type :• 0x00 – Terminal response • 0x01 – Event confirmation All other values are reserved. Indicates the action that the control point is expected to perform after receiving and processing the indication. If it is missing, the behavior described in Appendix C applies.
Type	0x69			1	Bearer Independent Protocol Status
Length	5			2	
Value	→	uint8	ch_id	1	Channel ID (see [S1], Section 8.7).

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum	status	4	Bearer Independent Protocol Status: • CAT_BIP_STATUS_IN_PROGRESS (0x00) – In progress • CAT_BIP_STATUS_END (0x01) – End All other values are reserved for future use and are to be ignored by the control point.
Type	0x6A			1	Refresh Alpha
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	pc_refresh_alpha_len	2	Number of sets of the following elements: • pc_refresh_alpha
		opaque	pc_refresh_alpha	Var	Refresh proactive command encoded as in [S1], Section 6.6.13. This is sent only if the refresh command contains alpha to be displayed.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_EVT_REGISTRATION_FAILED	Registration for one or more events failed, as it was registered by some other control point before
QMI_ERR_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_MISSING_ARG	One or more required TLVs are missing
QMI_ERR_OP_DEVICE_UNSUPPORTED	Device does not support the operation

3.2.4 Description of QMI_CAT_SET_EVENT_REPORT

The control point's event reporting state variables are modified according to the settings specified in the TLVs included in the request message. The service maintains a set of state variables for each control point.

Events of interest are communicated to the registered CAT control point via the QMI_CAT_EVENT_REPORT_IND message. A registration failure implies that none of the requested proactive commands in the QMI_CAT_SET_EVENT_REPORT_REQ are registered. The control point must register for these event report indications later, as required.

When the CAT control point is registered with the Refresh/Refresh Alpha bitmask (bit 11), different event TLV data are communicated to the control point based on the QMI_CAT configuration mode. If QMI_CAT is configured in Gobi mode, Refresh events with the refresh mode and refresh stage information are communicated to the registered control point. For all other QMI_CAT configuration modes, Refresh Alpha events with the alpha and icon (if any) are communicated to the registered control point.

When more than one card is available, applicable events are applied to all available cards, unless specified differently in the request.

The unsolicited indication message QMI_CAT_SET_EVENT_REPORT_IND is sent to interested control points when the device state corresponding to any TLV listed in Section 3.2 changes. Interested control points are those that previously registered successfully for the corresponding event to be reported using the QMI_CAT_SET_EVENT_REPORT_REQ message.

This indication message is generated when one or more corresponding proactive commands are received from the device. For certain proactive commands listed in Section 2.4.3, QMI_CAT includes a unique reference identifier. The control point uses the reference identifier while sending the terminal response for this indication. If no reference identifier is sent in the event report indication, the terminal response is not expected to be sent from the control point.

When the command is buffered, such as Setup Menu Event (as described in Section 2.4.3), the module sets uim_ref_id to 0xFFFF in the corresponding indication sent to the control point upon its registration. In this particular case, too, the control point is not expected to send a terminal response to this indication.

When QMI_CAT sends the decoded event indication to the control points, the TLV (0x1B) is mandatory in the message indication. See Appendix B for detailed information on mandatory or optional TLVs for each command.

Slot (0x50) is an optional TLV to indicate the slot ID for which the proactive command/event is being reported. If this TLV is missing, the control point assumes that the proactive command is coming from the card on slot 1.

3.3 QMI_CAT_GET_SUPPORTED_MSGS

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

CAT message ID

0x001E

Version introduced

Major - 1, Minor - 17

3.3.1 Request - QMI_CAT_GET_SUPPORTED_MSGS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.3.2 Response - QMI_CAT_GET_SUPPORTED_MSGS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Name	Common version introduced	Common version last modified
Result Code	1.6	1.7

Optional TLVs

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

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

Error codes

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

3.3.3 Description of QMI_CAT_GET_SUPPORTED_MSGS REQ/RESP

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

3.4 QMI_CAT_GET_SUPPORTED_FIELDS

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

CAT message ID

0x001F

Version introduced

Major - 1, Minor - 17

3.4.1 Request - QMI_CAT_GET_SUPPORTED_FIELDS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

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

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

Optional TLVs

None

3.4.2 Response - QMI_CAT_GET_SUPPORTED_FIELDS_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Name	Common version introduced	Common version last modified
Result Code	1.6	1.7

Optional TLVs

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

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

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

Error codes

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

3.4.3 Description of QMI_CAT_GET_SUPPORTED_FIELDS REQ/RESP

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

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

Examples are:

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

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

3.5 QMI_CAT_GET_SERVICE_STATE

Queries the QMI_CAT service state.

CAT message ID

0x0020

Version introduced

Major - 1, Minor - 0

3.5.1 Request - QMI_CAT_GET_SERVICE_STATE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.5.2 Response - QMI_CAT_GET_SERVICE_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. The following mandatory TLVs are present if the result code is QMI_RESULT_SUCCESS.

Name	Version introduced	Version last modified
CAT Service State	1.0	2.23

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	CAT Service State
Length	8			2	
Value	→	uint32	cat_common_evt_reg_state_mask	4	Bitmask of events registered by all control points: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh/Refresh Alpha (Refresh when QMI_CAT is configured in Gobi mode, Refresh Alpha in other cases) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Provide Local Information – Time • Bit 23 – Clients are to ignore this bit • Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Clients are to ignore this bit All unused bits are reserved for future use and are be ignored by the control point.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint32	pc_evt_report_mask	4	Bitmask of events registered by this control point: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh/Refresh Alpha (Refresh when QMI_CAT is configured in Gobi mode, Refresh Alpha in other cases) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Provide Local Information – Time • Bit 23 – Clients are to ignore this bit • Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Clients are to ignore this bit All unused bits are reserved for future use and are ignored by the control point.

Optional TLVs

The following TLVs are present if the result code is QMI_RESULT_SUCCESS in decoded format.

Name	Version introduced	Version last modified
Decoded CAT Service State	2.0	2.23
Full Function Event Service State	2.18	2.19

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Decoded CAT Service State
Length	8			2	
Value	→	uint32	cat_common_evt_reg_state_mask	4	Bitmask of decoded events registered by all control points: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh Alpha (not supported when QMI CAT is configured in Gobi mode) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Clients are to ignore this bit • Bit 23 – Smart Card Web Server • Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Bearer Independent Protocol Status All unused bits are reserved for future use and are ignored by the control point.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint32	pc_evt_report_mask	4	Bitmask of decoded events registered by this control point: • Bit 0 – Display Text • Bit 1 – Get Inkey • Bit 2 – Get Input • Bit 3 – Setup Menu • Bit 4 – Select Item • Bit 5 – Send SMS • Bit 6 – Setup Event – User Activity • Bit 7 – Setup Event – Idle Screen Notify • Bit 8 – Setup Event – Language Select Notify • Bit 9 – Setup Idle Mode Text • Bit 10 – Language Notification • Bit 11 – Refresh Alpha (not supported when QMI CAT is configured in Gobi mode) • Bit 12 – End Proactive Session • Bit 13 – Play Tone • Bit 14 – Setup Call • Bit 15 – Send DTMF • Bit 16 – Launch Browser • Bit 17 – Send SS • Bit 18 – Send USSD • Bit 19 – Provide Local Information – Language • Bit 20 – Bearer Independent Protocol • Bit 21 – Setup Event – Browser Termination • Bit 22 – Clients are to ignore this bit • Bit 23 – Smart Card Web Server • Bit 24 – Activate • Bit 25 – Setup Event – HCI connectivity • Bit 26 – Bearer Independent Protocol status All unused bits are reserved for future use and are ignored by the control point.
Type	0x11			1	Full Function Event Service State
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	mask32	pc_full_func_evt_report_mask	4	Full function event report bitmask registered by this control point: • Bit 0 – Send SMS • Bit 1 – Setup call • Bit 2 – Send DTMF • Bit 3 – Send SS • Bit 4 – Send USSD All unused bits are reserved for future use and are ignored by the control point.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_OP_DEVICE_UNSUPPORTED	Device does not support the operation
QMI_ERR_DEVICE_IN_USE	Device is currently in use

3.5.3 Description of QMI_CAT_GET_SERVICE_STATE REQ/RESP

This message retrieves the state of the QMI_CAT service.

The state includes the event report registration status of all control points of the service combined. In addition, the state also includes the event report registration status of this individual control point.

3.6 QMI_CAT_SEND_TR

Sends the terminal response to the proactive commands coming from the card.

CAT message ID

0x0021

Version introduced

Major - 1, Minor - 0

3.6.1 Request - QMI_CAT_SEND_TR_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Terminal Response	1.0	1.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Terminal Response
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID. This is the same reference ID as indicated in the event report indication for the relevant proactive command.
		uint16	tr_length	2	Number of sets of the following elements: • terminal_response
		opaque	terminal_response	Var	Terminal response for the relevant proactive command, encoded as in [S1], Section 6.8.

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.6.2 Response - QMI_CAT_SEND_TR_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

Name	Version introduced	Version last modified
TR Response	2.10	2.10

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	TR Response
Length	Var			2	
Value	→	uint8	sw1	1	Value of SW1 of the response, as defined in [S8] for ICC and [S9] for UICC.
		uint8	sw2	1	Value of SW2 of the response as defined in [S8] for ICC and [S9] for UICC.
		uint8	tr_response_len	1	Number of sets of the following elements: • tr_response
		opaque	tr_response	Var	TR response data.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_OPERATION	Invalid terminal response was requested to be sent to the card
QMI_ERR_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_INVALID_ARG	One of the TLVs in the message is invalid
QMI_ERR_OP_DEVICE_UNSUPPORTED	Device does not support the operation

3.6.3 Description of QMI_CAT_SEND_TR REQ/RESP

This message sends the terminal response as required by a received proactive command from the card. The terminal response must be encoded in the proper 3GPP format by the application as required for the given proactive command.

The terminal response is expected within a set time limit as defined by the target. After this expiry, the module sends a terminal response with the result code, unable to process command, to the card. Any subsequent terminal response issued by the control point after the expiry results in silent discarding of this response.

If the optional TLV for the slot is missing, the terminal response is sent by default, on slot 1.

3.7 QMI_CAT_SEND_ENVELOPE_CMD

Sends an envelope command to the card.

CAT message ID

0x0022

Version introduced

Major - 1, Minor - 0

3.7.1 Request - QMI_CAT_SEND_ENVELOPE_CMD_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Envelope Command	1.0	2.18

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Envelope Command
Length	Var			2	
Value	→	enum16	env_cmd_type	2	Envelope command type: • 0x01 – Menu Selection • 0x02 – Event DL User Activity • 0x03 – Event DL Idle Screen Available • 0x04 – Event DL Language Selection • 0x05 – Unknown Type • 0x06 – Event DL Browser Termination • 0x07 – Send Call Control • 0x08 – Event DL HCI Connectivity • 0x09 – SMS-PP Data Download • 0x0A – Event DL MT Call • 0x0B – Event DL Call Connected • 0x0C – Event DL Call Disconnected All other values are reserved.
		uint16	env_cmd_len	2	Number of sets of the following elements: • envelope_data
		opaque	envelope_data	Var	Encoded envelope response, as defined in [S1], Section 7.

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.7.2 Response - QMI_CAT_SEND_EVENLOPE_CMD_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

Name	Version introduced	Version last modified
Raw Envelope Respone Data	2.9	2.9

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Raw Envelope Respone Data
Length	Var			2	
Value	→	uint8	sw1	1	Value of SW1 of the response, as defined in [S8] for ICC and [S9] for UICC.
		uint8	sw2	1	Value of SW2 of the response, as defined in [S8] for ICC and [S9] for UICC.
		uint8	env_resp_data_len	1	Number of sets of the following elements: • env_resp_data
		opaque	env_resp_data	Var	Envelope response data.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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	One of the parameters specified contains an invalid value
QMI_ERR_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_CAT_INVALID_ENV_CMD	Invalid envelope command
QMI_ERR_CAT_ENV_CMD_BUSY	Card busy response for envelope command
QMI_ERR_CAT_ENV_CMD_FAIL	Envelope command failure
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.7.3 Description of QMI_CAT_SEND_ENVELOPE_CMD REQ/RESP

This message sends an envelope command, such as Menu Selection, to the card. The envelope command is triggered by the control point in response to one of the SIM proactive commands received previously, such as Setup Menu or Setup Event List, as described in Section 2.4.4. When the envelope response indicates that the card is busy, the control point tries to resend the envelope commands for event download, as in [S1], Section 7.5.

If the optional TLV for the slot is missing, the envelope command is sent by default, on slot 1.

3.8 QMI_CAT_GET_EVENT_REPORT

Retrieves the last proactive command from the modem.

CAT message ID

0x0023

Version introduced

Major - 2, Minor - 0

3.8.1 Request - QMI_CAT_GET_EVENT_REPORT_REQ

Message type

Request

Sender

Control Point

Mandatory TLVs

Name	Version introduced	Version last modified
Proactive Command Input	2.0	2.0

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Proactive Command Input
Length	5			2	
Value	→	uint32	cmd_ref_id	4	Command reference ID.
		enum8	format	1	Format in which to get the proactive command data: • 0x01 – Raw • 0x02 – Decoded

Optional TLVs

None

3.8.2 Response - QMI_CAT_GET_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

The following TLVs are optional. Some commands should expect to have certain TLVs present all the time.

Name	Version introduced	Version last modified
Display Text Event	1.0	1.0
Get Inkey Event	1.0	1.0
Get Input Event	1.0	1.0
Setup Menu Event	1.0	1.0
Select Item Event	1.0	1.0
Alpha Identifier Available	1.0	1.0
Setup Event List Event	1.0	1.0
Setup Idle Mode Text Event	1.0	1.0
Language Notification Event	1.0	1.0
Refresh Event	1.0	1.0
End Proactive Session	1.0	1.0
Decoded Header ID	2.0	2.23
Text String	2.0	2.0
High Priority	2.0	2.0
User Control	2.0	2.0
Icon	2.0	2.0
Duration	2.0	2.0
Response Format	2.0	2.0
Help Available	2.0	2.0
Response Packing Format	2.0	2.0
Response Length	2.0	2.0
Show User Input	2.0	2.0
Tone	2.0	2.9
Softkey Selection	2.0	2.0
Items	2.0	2.0
Default Item	2.0	2.0
Next Action Indicator	2.0	2.0
Icon ID List	2.0	2.12
Presentation	2.0	2.0
Packing Required	2.0	2.0
SMS TPDU	2.0	2.0
Is CDMA SMS	2.0	2.0

Name	Version introduced	Version last modified
Address	2.0	2.0
Call Setup Requirement	2.0	2.0
Redial	2.0	2.0
Subaddress	2.0	2.0
Capability Configuration	2.0	2.0
DTMF	2.0	2.0
Specific Language Notification	2.0	2.0
Language	2.0	2.0
Launch Mode	2.0	2.0
URL	2.0	2.0
Browser ID	2.0	2.0
Bearer List	2.0	2.0
Provisioning Files	2.0	2.0
USSD String	2.0	2.0
Default Text	2.0	2.0
Immediate Response Request	2.0	2.0
User Confirmation Alpha	2.0	2.0
Setup Call Display Alpha	2.0	2.0
User Confirmation Icon	2.0	2.0
Setup Call Display Icon	2.0	2.0
Gateway Proxy	2.0	2.0
Alpha	2.0	2.0
Notification Required	2.0	2.0
Play Tone Event	2.2	2.2
Setup Call Event	2.2	2.2
Send DTMF Event	2.2	2.2
Launch Browser Event	2.2	2.2
Send SMS Event	2.2	2.2
Send SS Event	2.2	2.2
Send USSD Event	2.2	2.2
Provide Local Information Event	2.2	2.2
Setup Event List Raw Event	2.2	2.2
Slot	2.2	2.20
Open Channel Event	2.3	2.3
Close Channel Event	2.3	2.3
Send Data Event	2.3	2.3
Receive Data Event	2.3	2.3
On Demand Link Establish	2.4	2.4
CSD Bearer Description	2.4	2.4
GPRS Bearer Description	2.4	2.4
EUTRAN External Parameter Bearer Description	2.4	2.4
EUTRAN External Mapped UTRAN PS Bearer Description	2.4	2.4
Buffer Size	2.4	2.4
Network Access Name	2.4	2.4
Other Address	2.4	2.4
User Login	2.4	2.4

Name	Version introduced	Version last modified
User Password	2.4	2.4
Transport Level	2.4	2.4
Data Destination Address	2.4	2.4
Channel Data Length	2.4	2.4
Send Data Immediately	2.4	2.4
Channel Data	2.4	2.4
Channel ID	2.4	2.4
Items with DCS	2.8	2.8
Activate Event	2.9	2.9
Activate Descriptor Target	2.9	2.9
Response Type	2.18	2.18
Bearer Independent Protocol Status	2.22	2.22
Refresh Alpha	2.23	2.23

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Display Text Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_display_text
		opaque	pc_display_text	Var	Display Text proactive command, encoded as in [S1], Section 6.6.1.
Type	0x11			1	Get Inkey Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_get_inkey
		opaque	pc_get_inkey	Var	Get Inkey proactive command, encoded as in [S1], Section 6.6.2.
Type	0x12			1	Get Input Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_get_input
		opaque	pc_get_input	Var	Get Input proactive command, encoded as in [S1], Section 6.6.3.
Type	0x13			1	Setup Menu Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_menu
		opaque	pc_setup_menu	Var	Setup Menu proactive command, encoded as in [S1], Section 6.6.7.

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x14			1	Select Item Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_select_item
		opaque	pc_select_item	Var	Select Item proactive command, encoded as in [S1], Section 6.6.8.
Type	0x15			1	Alpha Identifier Available (used only when QMI_CAT is configured in Gobi mode)
Length	Var			2	
Value	→	uint8	pc_cmd_type	1	Proactive command type that includes the alpha identifier: • 0x01 – Sends an SMS proactive command All other values are reserved.
		uint16	alpha_id_len	2	Number of sets of the following elements: • alpha_identifier
		opaque	alpha_identifier	Var	Alpha identifier, as in [S1], Section 8.2.
Type	0x16			1	Setup Event List Event (used only when QMI_CAT is configured in Gobi mode)
Length	4			2	
Value	→	uint32	pc_setup_evt_list	4	Setup event list bitmask: • Bit 0 – User Activity Notify • Bit 1 – Idle Screen Available • Bit 2 – Language Selection Notify Each set bit indicates the availability of the corresponding event in the Setup Event list proactive command. All unlisted bits are reserved for future use and are ignored.
Type	0x17			1	Setup Idle Mode Text Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_idle_mode_text
		opaque	pc_setup_idle_mode_text	Var	Setup Idle mode text proactive command, encoded as in [S1], Section 6.6.22.
Type	0x18			1	Language Notification Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint16	cmd_len	2	Number of sets of the following elements: • pc_lang_notification
		opaque	pc_lang_notification	Var	Language Notification proactive command, encoded as in [S1], Section 6.6.25.
Type	0x19			1	Refresh Event (used only when QMI_CAT is configured in Gobi mode)
Length	4			2	
Value	→	uint16	refresh_mode	2	As indicated in [S1], Section 8.6 (Command Qualifier for Refresh).
		enum16	refresh_stage	2	Stage of the refresh procedure: • 0x01 – Refresh start • 0x02 – Refresh success • 0x03 – Refresh failed
Type	0x1A			1	End Proactive Session
Length	1			2	
Value	→	enum8	proactive_session_end_type	1	Proactive session end type: • 0x01 – End proactive session command received from the card • 0x02 – End proactive session internal to the ME
Type	0x1B			1	Decoded Header ID
Length	6			2	
Value	→	enum8	command_id	1	ID of the proactive command: • 0x01 – Display Text • 0x02 – Get Inkey • 0x03 – Get Input • 0x04 – Launch Browser • 0x05 – Play Tone • 0x06 – Select Item • 0x07 – Send SMS • 0x08 – Send SS • 0x09 – Send USSD • 0x0A – Setup Call – User Confirmation • 0x0B – Setup Call – Alpha Display • 0x0C – Setup Menu • 0x0D – Setup Idle Text • 0x0E – Provide Local Information – Language • 0x0F – Send DTMF • 0x10 – Language Notification • 0x11 – Setup Event – User Activity • 0x12 – Setup Event – Idle Screen Notify

Field	Field value	Field type	Parameter	Size (byte)	Description
			command_id (cont.)		• 0x13 – Setup Event – Language Selection Notify • 0x14 – Open Channel • 0x15 – Close Channel • 0x16 – Receive Data • 0x17 – Send Data • 0x18 – Activate • 0x19 – Setup Event – HCI Connectivity • 0x1A – Refresh Alpha • 0X20 – Setup Event – Browser Termination All other values are reserved.
		uint32	uim_ref_id	4	Proactive command reference ID (used internally by the QMI_CAT service).
		uint8	command_number	1	Command number sent to the client in the proactive command for tracking purposes to match with the command number in the terminal response.
Type	0x1C			1	Text String
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x1D			1	High Priority
Length	1			2	
Value	→	enum8	high_priority	1	High priority value: • 0x00 – Do not clear the screen • 0x01 – Clear anything that is on the screen
Type	0x1E			1	User Control
Length	1			2	
Value	→	enum8	user_control	1	User control: • 0x00 – Do not allow the user to clear the screen • 0x01 – Allow the user to clear the screen
Type	0x1F			1	Icon
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color
		uint8	rec_num	1	Record number in the EF-IMG file.
		uint16	data_size	2	Number of sets of the following elements: • data
		opaque	data	Var	Image instance data in binary format.
Type	0x20			1	Duration
Length	2			2	
Value	→	enum8	units	1	Time units: • 0x00 – Minutes • 0x01 – Seconds • 0x02 – Tenths of seconds
		uint8	interval	1	Time interval; this number must be greater than zero (see [S1], Section 8.8).
Type	0x21			1	Response Format
Length	1			2	
Value	→	enum8	response_format	1	Response format: • 0x00 – SMS default alphabet • 0x01 – Yes/No • 0x02 – Numerical only • 0x03 – UCS2 • 0x04 – Immediate digit response • 0x05 – Yes/No and immediate digit response
Type	0x22			1	Help Available
Length	1			2	
Value	→	boolean	help_available	1	Whether help is available: • 0x00 – No help is available • 0x01 – Help is available
Type	0x23			1	Response Packing Format
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	response_packing_format	1	Response packing format: • 0x00 – Unpacked format • 0x01 – Packed format
Type	0x24			1	Response Length
Length	2			2	
Value	→	uint8	maximum_user_input	1	Maximum user input. A value of 0xFF indicates no maximum.
		uint8	minimum_user_input	1	Minimum user input. A value of 0x00 indicates no minimum.
Type	0x25			1	Show User Input
Length	1			2	
Value	→	enum8	show_user_input	1	Show user input: • 0x00 – ME can show * characters • 0x01 – ME can show user input
Type	0x26			1	Tone
Length	1			2	
Value	→	enum8	tone	1	Tone to be played: • 0x01 – Dial tone • 0x02 – Called subscriber busy tone • 0x03 – Congestion tone • 0x04 – Radio path ACK tone • 0x05 – Radio path not available, call drop tone • 0x06 – Error tone • 0x07 – Call waiting tone • 0x08 – Ringing tone • 0x09 – General beep • 0x0A – Positive ACK tone • 0x0B – Negative ACK tone • 0x0C – Ring tone selected by the user • 0x0D – SMS alert tone selected by the user • -1 – Not in use
Type	0x27			1	Softkey Selection
Length	1			2	
Value	→	enum8	softkey_selection	1	Softkey selection: • 0x00 – Softkey is not selected • 0x01 – Softkey is selected
Type	0x28			1	Items
Length	Var			2	
Value	→	uint8	number_of_items	1	Number of sets of the following elements: • item_id • item_text_length • item_text
		uint8	item_id	1	ID of the item. Each item has a unique identifier from 0x01 to 0xFF.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	item_text_length	1	Number of sets of the following elements: • item_text
		opaque	item_text	Var	Item text. Coded the same way that alpha is coded in the EF-ADN file (see [S6], clause 4.4.2.3).
Type	0x29			1	Default Item
Length	1			2	
Value	→	uint8	default_item	1	Default item to be selected. All values are valid, except 0xFF, which is reserved (see [S1], Section 8.10).
Type	0x2A			1	Next Action Indicator
Length	Var			2	
Value	→	uint8	num_of_items	1	Number of sets of the following elements: • next_action_list
		enum8	next_action_list	Var	Item in the action list: • 0x00 – Setup Call • 0x01 – Send SS • 0x02 – Send USSD • 0x03 – Send Short Message • 0x04 – Launch Browser • 0x05 – Play Tone • 0x06 – Display Text • 0x07 – Get Inkey • 0x08 – Get Input • 0x09 – Select Item • 0x0A – Setup Menu • 0x0B – Setup Idle Mode Text • 0x0C – End of the Proactive Session • 0x0D – Provide Local Information
Type	0x2B			1	Icon ID List
Length	Var			2	
Value	→	boolean	display_icon_only	1	Whether to display the icon only: • 0x00 – Icon is not self-explanatory, display icon with description • 0x01 – Icon is self-explanatory, display only the icon
		uint8	num_of_items	1	Number of sets of the following elements: • qualifier • height • width • ics • rec_num • data_size • data

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color
		uint8	rec_num	1	Record number in the EF-IMG file.
		uint16	data_size	2	Number of sets of the following elements: • data
		opaque	data	Var	Image instance data in binary format.
Type	0x2C			1	Presentation
Length	1			2	
Value	→	enum8	presentation	1	Presentation type: • 0x00 – Not specified • 0x01 – Data value presentation • 0x02 – Navigation presentation
Type	0x2D			1	Packing Required
Length	1			2	
Value	→	boolean	packing_required	1	Indicates whether packing is required: • 0x00 – Packing is not required • 0x01 – Packing is required
Type	0x2E			1	SMS TPDU
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • sms_tpdu
		opaque	sms_tpdu	Var	SMS TPDU data, as specified in [S6].
Type	0x2F			1	Is CDMA SMS
Length	1			2	
Value	→	boolean	is_cdma_sms	1	CDMA SMS format indication: • 0x00 – FALSE (3GPP format) • 0x01 – TRUE (3GPP2 format) This defaults to FALSE if the TLV is not present.
Type	0x30			1	Address
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x31			1	Call Setup Requirement
Length	1			2	
Value	→	enum8	call_setup_requirement	1	Call setup requirements: • 0x00 – No other calls • 0x01 – Hold active calls • 0x02 – Disconnect active calls
Type	0x32			1	Redial
Length	3			2	
Value	→	boolean	redial_necessary	1	Indicates whether redial is necessary: • 0x00 – Redial is not necessary • 0x01 – Redial is necessary
		enum8	units	1	Time units: • 0x00 – Minutes • 0x01 – Seconds • 0x02 – Tenths of seconds
		uint8	interval	1	Time interval. This value must be greater than zero if redial_necessary is set to 0x01 (see [S1], Section 8.8).
Type	0x33			1	Subaddress
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • subaddress
		opaque	subaddress	Var	Subaddress in BCD format (two digits encoded in one byte). Maximum size of the subaddress is 20 bytes (see [S1], Section 8.3).
Type	0x34			1	Capability Configuration
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint8	length	1	Number of sets of the following elements: • capability_config_data
		opaque	capability_config_data	Var	Capability configuration data (see [S1], Section 8.4).
Type	0x35			1	DTMF
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • dtmf_data
		opaque	dtmf_data	Var	DTMF data in BCD format (two digits encoded in one byte) (see [S1], Section 8.44).
Type	0x36			1	Specific Language Notification
Length	1			2	
Value	→	boolean	spec_lang_notify	1	Whether there is a specific language notification: • 0x00 – No • 0x01 – Yes
Type	0x37			1	Language
Length	2			2	
Value	→	uint16	language	2	Language value. Each language code is a pair of alphanumeric characters (defined in [S3]). Each alphanumeric character is coded on one byte using the SMS default 7-bit coded alphabet, as defined in [S1], Section 8.45, with bit 8 set to 0.
Type	0x38			1	Launch Mode
Length	1			2	
Value	→	enum8	launch_mode	1	Launch mode: • 0x00 – Launch if not already launched • 0x01 – Use the existing browser • 0x02 – Close the existing browser
Type	0x39			1	URL
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • url_data
		opaque	url_data	Var	URL (see [S1], Section 8.48).
Type	0x3A			1	Browser ID
Length	1			2	
Value	→	uint8	browser_id	1	Browser ID (see [S1], Section 8.47).
Type	0x3B			1	Bearer List
Length	Var			2	
Value	→	uint16	length	2	Number of sets of the following elements: • bearer_list

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum8	bearer_list	Var	Bearer list: • 0x00 – SMS • 0x01 – CSD • 0x02 – USSD bearer code • 0x03 – GPRS • 0x04 – Bearer default
Type	0x3C			1	Provisioning Files
Length	Var			2	
Value	→	uint32	num_of_prov_files	4	Number of sets of the following elements: • length • path
		uint8	length	1	Number of sets of the following elements: • path
		opaque	path	Var	Path to the provisioning file (see [S1], Section 8.50).
Type	0x3D			1	USSD String
Length	Var			2	
Value	→	enum8	orig_dcs_from_sim	1	Original data coding scheme from the SIM: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length	1	Number of sets of the following elements: • text
		opaque	text	Var	Text of USSD string (see [S4], Section 8.17).
Type	0x3E			1	Default Text
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x3F			1	Immediate Response Request
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	boolean	immediate_resp	1	Indicates whether an immediate response is required: • 0x00 – No • 0x01 – Yes
Type	0x40			1	User Confirmation Alpha
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x41			1	Setup Call Display Alpha
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x42			1	User Confirmation Icon
Length	Var			2	
Value	→	enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color
		uint8	rec_num	1	Record number in the EF-IMG file.
		uint16	data_size	2	Number of sets of the following elements: • data

Field	Field value	Field type	Parameter	Size (byte)	Description
		opaque	data	Var	Image instance data in binary format.
Type	0x43			1	Setup Call Display Icon
Length	Var			2	
Value	→	enum8	qualifier	1	Icon qualifier: • 0x00 – Icon is self-explanatory; it replaces the item text • 0x01 – Icon is not self-explanatory; it displays along with the text
		uint8	height	1	Icon height (from the EF-IMG file). Represents the number of raster image points.
		uint8	width	1	Icon width (from the EF-IMG file). Represents the number of raster image points.
		enum8	ics	1	Image coding scheme: • 0x00 – Unknown • 0x01 – Basic • 0x02 – Color
		uint8	rec_num	1	Record number in the EF-IMG file.
		uint16	data_size	2	Number of sets of the following elements: • data
		opaque	data	Var	Image instance data in binary format.
Type	0x44			1	Gateway Proxy
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x45			1	Alpha
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x46			1	Notification Required
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	boolean	notification_required	1	Indicates whether the notification for a setup event list is required: • 0 – Notification is not required • 1 – Notification is required
Type	0x47			1	Play Tone Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_play_tone
		opaque	pc_play_tone	Var	Play Tone proactive command, encoded as in [S1], Section 6.6.5.
Type	0x48			1	Setup Call Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_call
		opaque	pc_setup_call	Var	Setup Call proactive command, encoded as in [S1], Section 6.6.12.
Type	0x49			1	Send DTMF Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_dtmf
		opaque	pc_send_dtmf	Var	Send DTMF proactive command, encoded as in [S1], Section 6.6.24.
Type	0x4A			1	Launch Browser Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_launch_browser
		opaque	pc_launch_browser	Var	Launch Browser proactive command, encoded as in [S1], Section 6.6.26.
Type	0x4B			1	Send SMS Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_sms
		opaque	pc_send_sms	Var	Send SMS proactive command, encoded as in [S1], Section 6.6.9.
Type	0x4C			1	Send SS Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_ss
		opaque	pc_send_ss	Var	Send SS proactive command, encoded as in [S1], Section 6.6.10.
Type	0x4D			1	Send USSD Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_ussd
		opaque	pc_send_ussd	Var	Send USSD proactive command, encoded as in [S1], Section 6.6.11.
Type	0x4E			1	Provide Local Information Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_provide_local_info
		opaque	pc_provide_local_info	Var	Provide Local Information proactive command, encoded as in [S1], Section 6.6.15.
Type	0x4F			1	Setup Event List Raw Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_event_list
		opaque	pc_setup_event_list	Var	Setup Event List proactive command, encoded as in [S1], Section 6.6.16.
Type	0x50			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.
Type	0x51			1	Open Channel Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_open_channel
		opaque	pc_open_channel	Var	Open Channel proactive command, encoded as in [S1], Section 6.6.27.

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x52			1	Close Channel Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_close_channel
		opaque	pc_close_channel	Var	Close Channel proactive command, encoded as in [S1], Section 6.6.28.
Type	0x53			1	Send Data Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_send_data
		opaque	pc_send_data	Var	Send Data proactive command, encoded as in [S1], Section 6.6.30.
Type	0x54			1	Receive Data Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_receive_data
		opaque	pc_receive_data	Var	Receive Data proactive command, encoded as in [S1], Section 6.6.29.
Type	0x55			1	On Demand Link Establish
Length	1			2	
Value	→	boolean	on_demand_link_est	1	Indicates whether the link is required: • 0x00 – Link is not required • 0x01 – Link is required
Type	0x56			1	CSD Bearer Description
Length	3			2	
Value	→	uint8	speed	1	Data rate; same as the speed subparameter defined in [S5], Section 6.7.
		enum8	name	1	CSD bearer name: • 0x00 – Data Circuit Asynchronous; UDI or 3.1 kHz modem • 0x01 – Data Circuit Synchronous; UDI or 3.1 kHz modem • 0x02 – PAD Access Asynchronous UDI • 0x03 – Packet Access Synchronous UDI • 0x04 – Data Circuit Asynchronous RDI • 0x05 – Data Circuit Synchronous RDI • 0x06 – PAD Access Asynchronous RDI • 0x07 – Packet Access Synchronous RDI

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum8	connection_element	1	CSD bearer connection element: • 0x00 – Transparent • 0x01 – Nontransparent • 0x02 – Both, transparent preferred • 0x03 – Both, nontransparent preferred
Type	0x57			1	GPRS Bearer Description
Length	6			2	
Value	→	uint8	precedence_cls	1	Precedence class; same as the precedence subparameter defined in [S4], Section 8.52.2.
		uint8	delay_cls	1	Delay class; same as the delay subparameter defined in [S4], Section 8.52.2.
		uint8	reliability_cls	1	Reliability class; same as the reliability subparameter defined in [S4], Section 8.52.2.
		uint8	peak_throughput	1	Peak throughput class; same as the peak subparameter defined in [S4], Section 8.52.2.
		uint8	mean_throughput	1	Mean throughput class; same as the mean subparameter defined in [S4], Section 8.52.2.
		enum8	pkt_data_protocol	1	Packet Data Protocol: • 0x02 – IP All other values are reserved.
Type	0x58			1	EUTRAN External Parameter Bearer Description
Length	17			2	
Value	→	enum8	traffic_class	1	Indicates the type of application for which the UMTS bearer service is optimized: • 0x00 – Conversational • 0x01 – Streaming • 0x02 – Interactive • 0x03 – Background • 0x04 – Subscribed value All other values are reserved.
		uint16	max_bitrate_ul	2	Maximum bitrate UL; same as the maximum bitrate UL subparameter defined in [S4], Section 8.52.3.
		uint16	max_bitrate_dl	2	Maximum bitrate DL; same as the maximum bitrate DL subparameter defined in [S4], Section 8.52.3.
		uint16	guaranteed_bitrate_ul	2	Guaranteed bitrate UL; same as the guaranteed bitrate UL subparameter defined in [S4], Section 8.52.3.

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint16	guaranteed_bitrate_dl	2	Guaranteed bitrate DL; same as the guaranteed bitrate DL subparameter defined in [S4], Section 8.52.3.
		enum8	delivery_order	1	Numeric parameter that indicates if the UMTS bearer will provide in-sequence SDU delivery: • 0x00 – No • 0x01 – Yes • 0x02 – Subscribed value All other values are reserved.
		uint8	max_sdu_size	1	Maximum SDU size; same as the Maximum SDU size subparameter defined in [S4], Section 8.52.3.
		uint8	max_sdu_err_ratio	1	SDU error ratio; same as the SDU error ratio subparameter defined in [S4], Section 8.52.3.
		uint8	residual_bit_err_ratio	1	Residual bit error ratio; same as the residual bit error ratio subparameter defined in [S4], Section 8.52.3.
		enum8	delivery_of_err_sdu	1	Numeric parameter that indicates if SDUs detected as erroneous will be delivered: • 0x00 – No • 0x01 – Yes • 0x02 – No detect • 0x03 – Subscribed value All other values are reserved.
		uint8	transfer_delay	1	Transfer delay; same as the transfer delay subparameter defined in [S4], Section 8.52.3.
		uint8	traffic_handling_pri	1	Traffic handling priority; same as the traffic handling priority subparameter defined in [S4], Section 8.52.3.
		enum8	pdp_type	1	PDP type: • 0x02 – IP All other values are reserved.
Type	0x59			1	EUTRAN External Mapped UTRAN PS Bearer Description
Length	10			2	
Value	→	uint8	qci	1	QCI (see [S4], Section 8.52.5).
		uint8	max_bitrate_ul	1	Maximum bitrate UL (see [S4], Section 8.52.5).
		uint8	max_bitrate_dl	1	Maximum bitrate DL (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_ul	1	Guaranteed bitrate UL (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_dl	1	Guaranteed bitrate DL (see [S4], Section 8.52.5).

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	max_bitrate_ul_ext	1	Maximum bitrate UL Ext (see [S4], Section 8.52.5).
		uint8	max_bitrate_dl_ext	1	Maximum bitrate DL Ext (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_ul_ext	1	Guaranteed bitrate UL Ext (see [S4], Section 8.52.5).
		uint8	guaranteed_bitrate_dl_ext	1	Guaranteed bitrate DL Ext (see [S4], Section 8.52.5).
		enum8	pdp_type	1	PDP type: • 0x02 – IP All other values are reserved.
Type	0x5A			1	Buffer Size
Length	2			2	
Value	→	uint16	buffer_size	2	Buffer size.
Type	0x5B			1	Network Access Name
Length	Var			2	
Value	→	uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Network access name encoded in ASCII character (see [S4], Section 8.61).
Type	0x5C			1	Other Address
Length	Var			2	
Value	→	enum8	address_type	1	Address type: • 0x01 – No address given • 0x02 – Dynamic • 0x03 – IPv4 • 0x04 – IPv6 All other values are reserved.
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address (see [S1], Section 8.58).
Type	0x5D			1	User Login
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x5E			1	User Password
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x5F			1	Transport Level
Length	3			2	
Value	→	enum8	transport_protocol	1	Transport protocol: • 0x00 – Not present • 0x01 – UDP • 0x02 – TCP All other values are reserved.
		uint16	port_number	2	Port number.
Type	0x60			1	Data Destination Address
Length	Var			2	
Value	→	enum8	address_type	1	Address type: • 0x01 – No address given • 0x02 – Dynamic • 0x03 – IPv4 • 0x04 – IPv6 All other values are reserved.
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address (see [S1], Section 8.58).
Type	0x61			1	Channel Data Length
Length	1			2	
Value	→	uint8	ch_data_length	1	Number of bytes that are available in the channel buffer, or the number of bytes that are requested in a Received Data command (see [S1], Section 8.54).
Type	0x62			1	Send Data Immediately
Length	1			2	
Value	→	boolean	send_data_immediately	1	Indicates whether to send the data immediately: • 0x00 – No, store the data in the Tx buffer • 0x01 – Yes, send the data immediately
Type	0x63			1	Channel Data
Length	Var			2	
Value	→	uint16	data_len	2	Number of sets of the following elements: • channel_data_string

Field	Field value	Field type	Parameter	Size (byte)	Description
		opaque	channel_data_string	Var	Channel data string is considered by the terminal as binary coded on 8 bits (see [S1], Section 8.53).
Type	0x64			1	Channel ID
Length	1			2	
Value	→	uint8	ch_id	1	Channel ID (see [S1], Section 8.7).
Type	0x65			1	Items with DCS
Length	Var			2	
Value	→	uint8	number_of_items	1	Number of sets of the following elements: • item_id • dcs • item_text_length • item_text
		uint8	item_id	1	ID of the item. Each item has a unique identifier from 0x01 to 0xFF.
		enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	item_text_length	1	Number of sets of the following elements: • item_text
		opaque	item_text	Var	Item text (see [S6], clause 4.4.2.3).
Type	0x66			1	Activate Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	pc_activate_len	2	Number of sets of the following elements: • pc_activate
		opaque	pc_activate	Var	Activate proactive command encoded as in [S1], Section 6.6.40.
Type	0x67			1	Activate Descriptor Target
Length	1			2	
Value	→	enum8	target	1	Activate descriptor target (see [S1], Section 8.89): • 0x01 – UICC-CLF interface according to [S10] All other values are reserved for future use.
Type	0x68			1	Response Type
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum	rsp_type	4	Response type: • 0x00 – Terminal response • 0x01 – Event confirmation All other values are reserved. Indicates the action that the control point is expected to perform after receiving and processing the indication. If it is missing, the behavior described in Appendix C applies.
Type	0x69			1	Bearer Independent Protocol Status
Length	5			2	
Value	→	uint8	ch_id	1	Channel ID (see [S1], Section 8.7).
		enum	status	4	Bearer Independent Protocol Status: • CAT_BIP_STATUS_IN_PROGRESS (0x00) – In progress • CAT_BIP_STATUS_END (0x01) – End All other values are reserved for future use and are to be ignored by the control point.
Type	0x6A			1	Refresh Alpha
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	pc_refresh_alpha_len	2	Number of sets of the following elements: • pc_refresh_alpha
		opaque	pc_refresh_alpha	Var	Refresh proactive command encoded as in [S1], Section 6.6.13. This is sent only if the refresh command contains alpha to be displayed.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_CAT_INVALID_EVENT	Invalid event was requested to be sent to the card
QMI_ERR_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_OP_DEVICE_UNSUPPORTED	Device does not support the operation

3.8.3 Description of QMI_CAT_GET_EVENT_REPORT REQ/RESP

This message is called when the applications client knows that a modem event is pending. The client calls this message to tell the QMI_CAT to get the proactive command from the modem and return that data in the response.

If this command is being used to get indications from the modem, the QMI_CAT_EVENT_REPORT_IND command cannot be used.

When the QMI_CAT sends the decoded QMI_CAT_GET_EVENT_REPORT_RESP to control points, the TLV (0x1B) is mandatory in this message. See Appendix B for detailed information on mandatory TLVs and optional TLVs for each command in the decoded format.

If the optional TLV for the slot is missing, the control point assumes that the proactive command was received on slot 1.

3.9 QMI_CAT_SEND_DECODED_TR

Sends the Terminal Response (TR) in decoded format to the proactive commands coming from the card.

CAT message ID

0x0024

Version introduced

Major - 2, Minor - 0

3.9.1 Request - QMI_CAT_SEND_DECODED_TR_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Terminal Response	2.0	2.19

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Terminal Response
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID. This is the same reference ID as indicated in the event report indication for the relevant proactive command.
		uint8	command_number	1	Command number for which the terminal response is sent.

Field	Field value	Field type	Parameter	Size (byte)	Description
		enum8	response_cmd	1	Type of proactive command for which the terminal response is sent: • 0x01 – Display Text • 0x02 – Get Inkey • 0x03 – Get Input • 0x04 – Launch Browser • 0x05 – Play Tone • 0x06 – Select Item Request • 0x07 – Setup Menu • 0x08 – Setup Idle Text • 0x09 – Provide Local Information – Language • 0x0A – Setup Event – User Activity • 0x0B – Setup Event – Idle Screen Notify • 0x0C – Setup Event – Language Select Notify • 0x0D – Language Notification • 0x0E – Activate • 0x0F – Setup Event – HCI Connectivity • 0x10 – Setup Event – Browser Termination • 0x11 – Send SMS • 0x12 – Setup Call • 0x13 – Send DTMF • 0x14 – Send SS • 0x15 – Send USSD All other values are reserved.
		enum8	general_result	1	Result of the proactive command, as defined in [S1], Section 8.12.
		uint8	additional_info_length	1	Number of sets of the following elements: • tr_additional_info
		opaque	tr_additional_info	Var	Additional information is only required for some commands. [S1], Section 8.12, describes the additional information. The maximum size is 10.

Optional TLVs

Name	Version introduced	Version last modified
Text String	2.0	2.0
Item Identifier	2.0	2.0
Get Inkey Extra Info	2.0	2.8
Language Info	2.1	2.1
Slot	2.2	2.20
Get Inkey Yes/No Info	2.8	2.8

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Text String
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x11			1	Item Identifier
Length	1			2	
Value	→	uint8	identifier	1	Identifier of the item chosen: • 0x00 – NULL identifier • 0x01 to 0xFF – Value of the item
Type	0x12			1	Get Inkey Extra Info
Length	Var			2	
Value	→	enum8	unit	1	Time units: • 0x00 – Minutes • 0x01 – Seconds • 0x02 – Tenths of seconds • -1 – Duration is not present
		uint8	interval	1	Time interval. This number must be greater than zero.
		enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x13			1	Language Info
Length	2			2	
Value	→	uint16	language	2	Language value. Each language code is a pair of alphanumeric characters (defined in [S3]). Each alphanumeric character is coded on one byte using the SMS default 7-bit coded alphabet, as defined in [S1], Section 8.45, with bit 8 set to 0.
Type	0x14			1	Slot
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.
Type	0x15			1	Get Inkey Yes/No Info
Length	3			2	
Value	→	enum8	unit	1	Time units: • 0x00 – Minutes • 0x01 – Seconds • 0x02 – Tenths of seconds • -1 – Duration is not present
		uint8	interval	1	Time interval. This number must be greater than zero.
		enum8	get_inkey_yes_no	1	Yes/No input for get inkey: • 0x00 – No • 0x01 – Yes If a text input is required from the user, the Get Inkey Extra Info TLV must be used.

3.9.2 Response - QMI_CAT_SEND_DECODED_TR_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

Name	Version introduced	Version last modified
TR Response	2.10	2.10

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	TR Response
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint8	sw1	1	Value of SW1 of the response, as defined in [S8] for ICC and [S9] for UICC.
		uint8	sw2	1	Value of SW2 of the response as defined in [S8] for ICC and [S9] for UICC.
		uint8	tr_response_len	1	Number of sets of the following elements: • tr_response
		opaque	tr_response	Var	TR response data.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_OPERATION	Invalid terminal response was requested to be sent to the card
QMI_ERR_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_INVALID_ARG	One of the TLVs in the message is invalid
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.9.3 Description of QMI_CAT_SEND_DECODED_TR REQ/RESP

This message sends the terminal response, as required by a received proactive command from the card.

The terminal response is expected within a set time limit, as defined by the target. After this expiry, the module sends a terminal response with the result code, unable to process command, to the card. Any subsequent terminal response issued by the control point after the expiry results in silent discarding of this response.

If the optional TLV for the slot is missing, the terminal response is sent by default on slot 1.

3.10 QMI_CAT_SEND_DECODED_ENVELOPE_CMD

Sends an envelope command in decoded format to the card.

CAT message ID

0x0025

Version introduced

Major - 2, Minor - 0

3.10.1 Request - QMI_CAT_SEND_DECODED_ENVELOPE_CMD_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Envelope Command	2.0	2.19

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Envelope Command
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	env_cmd_type	1	Decoded envelope command type. See Appendix D for information on mandatory and optional TLVs for each envelope command. • 0x01 – Menu Selection • 0x02 – Event DL Language Selection • 0x03 – Event DL User Activity • 0x04 – Event DL Idle Screen Available • 0x05 – Send Call Control • 0x06 – Event DL HCI Connectivity • 0x07 – Event DL Browser Termination • 0x08 – SMS-PP Data Download • 0x09 – Event DL MT Call • 0x0A – Event DL MT Call Connected • 0x0B – Event DL MO Call Connected • 0x0C – Event DL Call Disconnected near end • 0x0D – Event DL Call Disconnected far end All other values are reserved.

Optional TLVs

Name	Version introduced	Version last modified
Item Identifier	2.0	2.0
Help Request	2.0	2.0
Language	2.0	2.0
Slot	2.2	2.20
Address	2.5	2.5
Subaddress	2.5	2.5
Capability Configuration Parameter 1	2.5	2.5
Capability Configuration Parameter 2	2.5	2.5
USSD String	2.5	2.5
PDP Context Activation	2.5	2.5
EPS PDN Connect Activation	2.5	2.5
Browser Termination Cause	2.12	2.12
SMS TPDU	2.15	2.15
Is CDMA SMS	2.16	2.16
Radio Access Technology	2.18	2.25
Call Type	2.18	2.25
Transaction ID	2.18	2.18
RP Address	2.18	2.18
TP Address	2.18	2.18
Cause	2.18	2.18
IMS Request - URI	2.24	2.24

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Item Identifier
Length	1			2	
Value	→	uint8	identifier	1	Identifier of the item chosen.
Type	0x11			1	Help Request
Length	1			2	
Value	→	boolean	help_request	1	Whether help is requested: • 0x00 – No help is requested • 0x01 – Help is requested
Type	0x12			1	Language
Length	2			2	
Value	→	uint16	language	2	Language value. Each language code is a pair of alphanumeric characters (defined in [S3]). Each alphanumeric character is coded on one byte using the SMS default 7-bit coded alphabet, as defined in [S1], Section 8.45, with bit 8 set to 0.
Type	0x13			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.
Type	0x14			1	Address
Length	Var			2	
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x15			1	Subaddress
Length	Var			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	uint8	length	1	Number of sets of the following elements: • subaddress
		opaque	subaddress	Var	Subaddress in BCD format (two digits encoded in one byte). Maximum size of the subaddress is 20 bytes (see [S1], Section 8.3).
Type	0x16			1	Capability Configuration Parameter 1
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • capability_config_data
		opaque	capability_config_data	Var	Capability configuration data (see [S1], Section 8.4).
Type	0x17			1	Capability Configuration Parameter 2
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • capability_config_data
		opaque	capability_config_data	Var	Capability configuration data (see [S1], Section 8.4).
Type	0x18			1	USSD String
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x19			1	PDP Context Activation
Length	Var			2	
Value	→	uint8	pdp_context_act_data_len	1	Number of sets of the following elements: • pdp_context_act_data
		opaque	pdp_context_act_data	Var	PDP context activation data. Coded as the Activate PDP Context Request message, specified in [S6].
Type	0x1A			1	EPS PDN Connect Activation
Length	Var			2	
Value	→	uint8	eps_pdn_connect_act_data_len	1	Number of sets of the following elements: • eps_pdn_connect_act_data

Field	Field value	Field type	Parameter	Size (byte)	Description
		opaque	eps_pdn_connect_act_data	Var	EPS PDN connect activation data; coded as the PDN Connectivity Request message, specified in [S7].
Type	0x1B			1	Browser Termination Cause
Length	4			2	
Value	→	enum	browser_term_cause	4	Browser termination cause: • 0x00000000 – CAT_BROWSER_TERM_CAUSE_TYPE_USER_TERMINATED – User terminated the browser • 0x00000001 – CAT_BROWSER_TERM_CAUSE_TYPE_ERROR – Browser terminated due to error
Type	0x1C			1	SMS TPDU
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • sms_tpdu
		opaque	sms_tpdu	Var	SMS TPDU data, as specified in [S6].
Type	0x1D			1	Is CDMA SMS
Length	1			2	
Value	→	boolean	is_cdma_sms	1	CDMA SMS format indication: • 0x00 – FALSE (3GPP format) • 0x01 – TRUE (3GPP2 format) This defaults to FALSE if the TLV is not present.
Type	0x1E			1	Radio Access Technology
Length	4			2	
Value	→	enum	rat	4	Access technology type: • 0x00000000 – CAT_ACCESS_TECH_NONE – RAT is unknown • 0x00000001 – CAT_ACCESS_TECH_GSM – GSM is used • 0x00000002 – CAT_ACCESS_TECH_UTRAN – UTRAN is used • 0x00000003 – CAT_ACCESS_TECH_CDMA – CDMA is used • 0x00000004 – CAT_ACCESS_TECH_LTE – LTE is used • 0x00000005 – CAT_ACCESS_TECH_WLAN – WLAN is used
Type	0x1F			1	Call Type
Length	4			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum	call_type	4	Call Type: • 0x00000000 – CAT_VOICE – Voice • 0x00000001 – CAT_SS – SS • 0x00000002 – CAT_USSD – USSD • 0x00000003 – CAT_SMS – SMS • 0x00000004 – CAT_IMS – IMS
Type	0x20			1	Transaction ID
Length	Var			2	
Value	→	uint8	transaction_id_len	1	Number of sets of the following elements: • transaction_id
		opaque	transaction_id	Var	Call transaction ID (see [S1], Section 8.28).
Type	0x21			1	RP Address
Length	Var			2	
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x22			1	TP Address
Length	Var			2	
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x23			1	Cause
Length	Var			2	
Value	→	uint8	cause_len	1	Number of sets of the following elements: • cause
		opaque	cause	Var	Cause (see [S1], Section 8.26).
Type	0x24			1	IMS Request - URI
Length	Var			2	
Value	→	uint8	ims_request_uri_len	1	Number of sets of the following elements: • ims_request_uri
		char	ims_request_uri	Var	

3.10.2 Response - QMI_CAT_SEND_DECODED_ENVELOPE_CMD_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

Name	Version introduced	Version last modified
Call Control Result	2.5	2.5
Address	2.5	2.5
Subaddress	2.5	2.5
Capability Configuration Parameter 1	2.5	2.5
Capability Configuration Parameter 2	2.5	2.5
USSD String	2.5	2.5
PDP Context Activation	2.5	2.5
EPS PDN Connect Activation	2.5	2.5
Alpha	2.5	2.5
BC Repeat Indicator	2.5	2.5
SMS-PP Data Download UICC Acknowledgment	2.15	2.15

Name	Version introduced	Version last modified
RP Address	2.18	2.18
TP Address	2.18	2.18
IMS Request - URI	2.24	2.24

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Call Control Result
Length	1			2	
Value	→	enum8	cc_result	1	Call control result: • 0x00 – Call control result is allowed with no modification • 0x01 – Call control result is not allowed • 0x02 – Call control result is allowed with modification
Type	0x11			1	Address
Length	Var			2	
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x12			1	Subaddress
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • subaddress
		opaque	subaddress	Var	Subaddress in BCD format (two digits encoded in one byte). Maximum size of the subaddress is 20 bytes (see [S1], Section 8.3).
Type	0x13			1	Capability Configuration Parameter 1
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • capability_config_data

Field	Field value	Field type	Parameter	Size (byte)	Description
		opaque	capability_config_data	Var	Capability configuration data (see [S1], Section 8.4).
Type	0x14			1	Capability Configuration Parameter 2
Length	Var			2	
Value	→	uint8	length	1	Number of sets of the following elements: • capability_config_data
		opaque	capability_config_data	Var	Capability configuration data (see [S1], Section 8.4).
Type	0x15			1	USSD String
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.
Type	0x16			1	PDP Context Activation
Length	Var			2	
Value	→	uint8	pdp_context_act_data_len	1	Number of sets of the following elements: • pdp_context_act_data
		opaque	pdp_context_act_data	Var	PDP context activation data. Coded as the Activate PDP Context Request message, specified in [S6].
Type	0x17			1	EPS PDN Connect Activation
Length	Var			2	
Value	→	uint8	eps_pdn_connect_act_data_len	1	Number of sets of the following elements: • eps_pdn_connect_act_data
		opaque	eps_pdn_connect_act_data	Var	EPS PDN connect activation data; coded as the PDN Connectivity Request message, specified in [S7].
Type	0x18			1	Alpha
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x19			1	BC Repeat Indicator
Length	1			2	
Value	→	enum8	bc_repeat_ind	1	Bearer capability repeat indicator: • 0x00 – Alternate mode • 0x01 – Sequential mode
Type	0x1A			1	SMS-PP Data Download UICC Acknowledgment
Length	Var			2	
Value	→	uint8	sms_pp_uicc_acknowledge_len	1	Number of sets of the following elements: • sms_pp_uicc_acknowledge
		opaque	sms_pp_uicc_acknowledge	Var	SMS-PP data download envelope response, as defined in [S4], Section 7.1.
Type	0x1B			1	RP Address
Length	Var			2	
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x1C			1	TP Address
Length	Var			2	
Value	→	enum8	ton	1	TON of the address: • 0x00 – Unknown • 0x01 – International number • 0x02 – National number • 0x03 – Network-specific number
		enum8	npi	1	NPI of the address: • 0x00 – Unknown • 0x01 – ISDN telephony • 0x02 – Data NPI • 0x03 – Telex NPI • 0x04 – Private NPI • 0x0F – Extension is reserved

Field	Field value	Field type	Parameter	Size (byte)	Description
		uint8	length	1	Number of sets of the following elements: • address_data
		opaque	address_data	Var	Address in byte-based BCD format. The maximum length of the address is 200 bytes (see [S1], Section 8.1).
Type	0x1D			1	IMS Request - URI
Length	Var			2	
Value	→	uint8	ims_request_uri_len	1	Number of sets of the following elements: • ims_request_uri
		char	ims_request_uri	Var	

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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	One of the parameters specified contains an invalid value
QMI_ERR_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_CAT_INVALID_ENV_CMD	Invalid envelope command
QMI_ERR_CAT_ENV_CMD_BUSY	Card busy response for envelope command
QMI_ERR_CAT_ENV_CMD_FAIL	Envelope command failure
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support the operation
QMI_ERR_INVALID_OPERATION	Operation performed by the client was not carried out; see the Note below

3.10.3 Description of QMI_CAT_SEND_DECODED_ENVELOPE_CMD REQ/RESP

This message sends an envelope command, such as Menu Selection, to the card. When the envelope response indicates the card is busy, the control point retries sending the envelope commands for event download as in [S1], Section 7.5.

If the optional TLV for the slot is missing, the envelope command is sent by default on slot 1.

See Appendix D for information about mandatory and optional TLVs for envelope commands that apply to QMI_CAT_SEND_DECODED_ENVELOPE_CMD_REQ.

3.11 QMI_CAT_EVENT_CONFIRMATION

Sends user and icon confirmation for network-related commands.

CAT message ID

0x0026

Version introduced

Major - 2, Minor - 0

3.11.1 Request - QMI_CAT_EVENT_CONFIRMATION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
User Confirmed	2.0	2.0
Icon is Displayed	2.0	2.0
Slot	2.2	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	User Confirmed
Length	1			2	
Value	→	boolean	confirm	1	User confirmed: • 0x00 – No • 0x01 – Yes
Type	0x11			1	Icon is Displayed
Length	1			2	
Value	→	boolean	display	1	Icon is displayed: • 0x00 – No • 0x01 – Yes
Type	0x12			1	Slot
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.11.2 Response - QMI_CAT_EVENT_CONFIRMATION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_ARG_TOO_LONG	One of the TLVs in the message too long
QMI_ERR_INVALID_OPERATION	Invalid terminal response was requested to be sent to the card
QMI_ERR_INVALID_ARG	One of the TLVs in the message is invalid
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.11.3 Description of QMI_CAT_EVENT_CONFIRMATION REQ/RESP

This message sends user and/or icon confirmation as required by a received network-related proactive command (Setup Call, Send SMS, Send SS, Send DTMF, Send USSD) from the card.

The User Confirmed TLV is used only for the SETUP CALL and OPEN CHANNEL commands from the card for the user confirmation phase.

The Icon is Displayed TLV is used only when the proactive command contains an icon.

The application invokes this command after any network-related proactive command from the card, even if user confirmation and icon confirmation are not required, in that case passing an empty payload.

If the optional TLV for the slot is missing, the confirmation is sent by default on slot 1.

3.12 QMI_CAT_SCWS_OPEN_CHANNEL

Sends the Open Channel indication to the Smart Card Web Server (SCWS) agent and indicates a QMI_CAT event.

CAT message ID

0x0027

Version introduced

Major - 2, Minor - 6

3.12.1 Request - QMI_CAT_SCWS_OPEN_CHANNEL_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Channel Status	2.6	2.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Channel Status
Length	5			2	
Value	→	uint32	ch_id	4	Channel ID.
		enum8	state	1	Channel state: • 0x00 – Closed state • 0x01 – Listen state • 0x02 – Established state Other values are reserved for future use.

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.6	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.12.2 Response - QMI_CAT_SCWS_OPEN_CHANNEL_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

3.12.3 Indication - QMI_CAT_SCWS_OPEN_CHANNEL_IND

Message type

Indication

Sender

Service

Scope

Unicast (per control point)

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
Open Channel Information	2.6	2.6
Slot	2.6	2.20
Alpha	2.13	2.13

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Open Channel Information
Length	8			2	
Value	→	uint32	ch_id	4	Channel ID to be used for the SCWS connection.
		uint16	port	2	Port for the local TCP socket.
		uint16	buffer_size	2	Buffer size to be used.
Type	0x11			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.
Type	0x12			1	Alpha
Length	Var			2	
Value	→	enum8	dcs	1	Data coding scheme: • 0x00 – 7-bit GSM • 0x01 – 8-bit GSM • 0x02 – UCS2
		uint8	length_of_string	1	Number of sets of the following elements: • text
		opaque	text	Var	Text string data in the specified data coding scheme.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.12.4 Description of QMI_CAT_SCWS_OPEN_CHANNEL

The QMI_CAT_SCWS_OPEN_CHANNEL_REQ request is sent to the service as a response for the QMI_CAT_SCWS_OPEN_CHANNEL_IND after the SCWS agent opens the required TCP socket. The Listen state indicates a success, and the Closed state indicates a failure. Other values are reserved for future use.

The unsolicited indication message QMI_CAT_SCWS_OPEN_CHANNEL_IND is sent to the control point when a new OPEN CHANNEL proactive command is received from the SIM card for the Smart Card Web Server functionality. At this point, the SCWS agent must open a TCP socket and then invoke QMI_CAT_SCWS_OPEN_CHANNEL_REQ with the state of the socket to indicate the result of the operation to the modem.

3.13 QMI_CAT_SCWS_CLOSE_CHANNEL

Sends the Close Channel indication to the SCWS agent and indicates a QMI_CAT event.

CAT message ID

0x0028

Version introduced

Major - 2, Minor - 6

3.13.1 Request - QMI_CAT_SCWS_CLOSE_CHANNEL_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Channel Status	2.6	2.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Channel Status
Length	5			2	
Value	→	uint32	ch_id	4	Channel ID.
		enum8	state	1	Channel state: • 0x00 – Closed state • 0x01 – Listen state • 0x02 – Established state Other values are reserved for future use.

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.13.2 Response - QMI_CAT_SCWS_CLOSE_CHANNEL_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

3.13.3 Indication - QMI_CAT_SCWS_CLOSE_CHANNEL_IND

Message type

Indication

Sender

Service

Scope

Unicast (per control point)

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
Close Channel Information	2.6	2.6
Slot	2.6	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Close Channel Information
Length	5			2	
Value	→	uint32	ch_id	4	Channel ID to be used for the SCWS connection.
		enum8	state	1	Channel state: • 0x00 – Closed state; indicates that the socket must be closed • 0x01 – Listen state; indicates that the client needs to be disconnected; the socket remains open in the Listen state • 0x02 – Established state Other values are reserved for future use.
Type	0x11			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_OP_DEVICE_UNSUPPORTED	Device does not support the operation

3.13.4 Description of QMI_CAT_SCWS_CLOSE_CHANNEL

The QMI_CAT_SCWS_CLOSE_CHANNEL_REQ request is sent to the service as a response for the QMI_CAT_SCWS_CLOSE_CHANNEL_IND after the SCWS agent closes the required TCP socket, as per the state indicated in the indication.

The state in the request indicates the new state of the socket; in case of success, it will have the same value as in the corresponding indication.

The unsolicited indication message QMI_CAT_SCWS_CLOSE_CHANNEL_IND is sent to the control point when a new CLOSE CHANNEL proactive command is received from the SIM card for the SCWS functionality. The SCWS agent then closes or disconnects the corresponding socket, depending on the state indicated in the indication, and invokes QMI_CAT_SCWS_CLOSE_CHANNEL_REQ to confirm the status of the socket.

3.14 QMI_CAT_SCWS_SEND_DATA

Sends data to the SCWS agent and indicates a QMI_CAT event.

CAT message ID

0x0029

Version introduced

Major - 2, Minor - 6

3.14.1 Request - QMI_CAT_SCWS_SEND_DATA_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Channel Status	2.6	2.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Channel Status
Length	5			2	
Value	→	uint32	ch_id	4	Channel ID
		boolean	result	1	Result of the Send Data command: • 0x00 – Failed • 0x01 – Success

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.14.2 Response - QMI_CAT_SCWS_SEND_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

3.14.3 Indication - QMI_CAT_SCWS_SEND_DATA_IND

Message type

Indication

Sender

Service

Scope

Unicast (per control point)

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
Send Data Information	2.6	2.6
Slot	2.6	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Send Data Information
Length	Var			2	
Value	→	uint32	ch_id	4	Channel ID to be used to send the data.
		uint8	total_packets	1	Total number of packets.
		uint8	current_packet	1	Current packet.
		uint16	data_len	2	Number of sets of the following elements: • data
		opaque	data	Var	Data to be sent.
Type	0x11			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.14.4 Description of QMI_CAT_SCWS_SEND_DATA

The QMI_CAT_SCWS_SEND_DATA_REQ request is sent to the service as a response for the QMI_CAT_SCWS_SEND_DATA_IND after the SCWS agent sends the data on the TCP socket, to indicate the result of the operation.

The indication message QMI_CAT_SCWS_SEND_DATA_IND is sent to the control point with buffered data from the SIM card. The SCWS agent must send the data using the corresponding TCP socket. The control point executes QMI_CAT_SCWS_SEND_DATA_REQ, indicating the result of the operation only after the total number of packets is completed.

Due to the size limitation of the QMI, the data might be fragmented by the modem if it is greater than the amount of data that can be transferred at one time through the QMI.

3.15 QMI_CAT_SCWS_DATA_AVAILABLE

Indicates that data is available.

CAT message ID

0x002A

Version introduced

Major - 2, Minor - 6

3.15.1 Request - QMI_CAT_SCWS_DATA_AVAILABLE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Remaining Data	2.6	2.6
Length of the Remaining Data	2.6	2.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Remaining Data
Length	Var			2	
Value	→	uint32	ch_id	4	Channel ID.
		uint16	data_len	2	Number of sets of the following elements: • data
		opaque	data	Var	Data that is received.
Type	0x02			1	Length of the Remaining Data
Length	2			2	
Value	→	uint16	remaining_data_len	2	Remaining data length.

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.15.2 Response - QMI_CAT_SCWS_DATA_AVAILABLEA_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.15.3 Description of QMI_CAT_SCWS_DATA_AVAILABLE REQ/RESP

This message is sent by the Control Point when data is written to the socket. This message informs the modem that the data is available to be sent to the card.

The request allows the SCWS agent to fragment the data in small chunks to be sent over the QMI interface. The data is sent to the card only when the last fragment is received with the remaining data length indicating zero. All fragments are sent in order by the SCWS agent.

3.16 QMI_CAT_SCWS_CHANNEL_STATUS

Informs the modem about a change in the channel state.

CAT message ID

0x002B

Version introduced

Major - 2, Minor - 6

3.16.1 Request - QMI_CAT_SCWS_CHANNEL_STATUS_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Channel Status	2.6	2.6

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Channel Status
Length	5			2	
Value	→	uint32	ch_id	4	Channel ID.
		enum8	state	1	Channel state: • 0x00 – Closed state • 0x01 – Listen state • 0x02 – Established state Other values are reserved for future use.

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	

Field	Field value	Field type	Parameter	Size (byte)	Description
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.16.2 Response - QMI_CAT_SCWS_CHANNEL_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

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_ARG_TOO_LONG	One of the TLVs in the message is too long
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.16.3 Description of QMI_CAT_SCWS_CHANNEL_STATUS REQ/RESP

This message is sent by the SCWS agent when there is a change in the channel status.

3.17 QMI_CAT_GET_TERMINAL_PROFILE

Retrieves the current modem terminal profile.

CAT message ID

0x002C

Version introduced

Major - 2, Minor - 10

3.17.1 Request - QMI_CAT_GET_TERMINAL_PROFILE_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.10	2.20

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.17.2 Response - QMI_CAT_GET_TERMINAL_PROFILE_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Optional TLVs

Name	Version introduced	Version last modified
Raw Terminal Profile Data	2.10	2.10

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Raw Terminal Profile Data
Length	Var			2	
Value	→	uint8	terminal_profile_data_len	1	Number of sets of the following elements: • terminal_profile_data
		opaque	terminal_profile_data	Var	Terminal profile data.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An unexpected error occurred during processing
QMI_ERR_MALFORMED_MSG	Message was not formulated correctly by the control point or the message was corrupted during transmission
QMI_ERR_NO_MEMORY	Device could not allocate memory to formulate a response
QMI_ERR_DEVICE_NOT_READY	Device is yet not ready to process the request
QMI_ERR_OP_DEVICE_UNSUPPORTED	Device does not support the operation

3.17.3 Description of QMI_CAT_GET_TERMINAL_PROFILE REQ/RESP

This message is sent by the Control Point to retrieve the current terminal profile.

If the optional TLV for the slot is missing, the control point assumes that the request is for the card on slot 1.

3.18 QMI_CAT_SET_CONFIGURATION

Changes the configuration of the QMI_CAT service.

CAT message ID

0x002D

Version introduced

Major - 2, Minor - 11

3.18.1 Request - QMI_CAT_SET_CONFIGURATION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

Name	Version introduced	Version last modified
Configuration Mode	2.11	2.11

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	Configuration Mode
Length	1			2	
Value	→	enum8	cat_config_mode	1	QMI_CAT configuration mode: • 0x00 – Disabled mode • 0x01 – Gobi mode • 0x02 – Android mode • 0x03 – Decoded mode • 0x04 – Decoded Pull-only mode • 0x05 – Custom Raw mode (allows a customizable terminal profile for raw format) • 0x06 – Custom Decoded mode (allows a customizable terminal profile for decoded format) Other values are reserved for future use.

Optional TLVs

Name	Version introduced	Version last modified
Custom Terminal Profile Data	2.11	2.11

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Custom Terminal Profile Data
Length	Var			2	
Value	→	uint8	custom_tp_len	1	Number of sets of the following elements: • custom_tp
		opaque	custom_tp	Var	Custom terminal profile, encoded as in [S1], Section 5.2. The first byte of the TP bitmask starts from custom_tp[0]. This TLV is used only for custom modes and ignored in all other cases.

3.18.2 Response - QMI_CAT_SET_CONFIGURATION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Name	Version introduced	Version last modified
Result Code	Unknown	2.11

Optional TLVs

None

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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_EFFECT	Operation had no effect

3.18.3 Description of QMI_CAT_SET_CONFIGURATION REQ/RESP

This message is used to set the configuration mode of the QMI_CAT service. The new configuration might only take effect after a reboot. The client can verify if the new configuration is in use using the QMI_CAT_GET_CONFIGURATION request.

3.19 QMI_CAT_GET_CONFIGURATION

Gets the configuration of the QMI_CAT service.

CAT message ID

0x002E

Version introduced

Major - 2, Minor - 11

3.19.1 Request - QMI_CAT_GET_CONFIGURATION_REQ

Message type

Request

Sender

Control point

Mandatory TLVs

None

Optional TLVs

None

3.19.2 Response - QMI_CAT_GET_CONFIGURATION_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Name	Version introduced	Version last modified
Result Code	Unknown	2.11

Optional TLVs

Name	Version introduced	Version last modified
Configuration Mode	2.11	2.11
Custom Terminal Profile Data	2.11	2.11

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Configuration Mode
Length	1			2	
Value	→	enum8	cat_config_mode	1	Current QMI_CAT configuration mode: • 0x00 – Disabled mode • 0x01 – Gobi mode • 0x02 – Android mode • 0x03 – Decoded mode • 0x04 – Decoded Pull-only mode • 0x05 – Custom Raw mode (allows a customizable terminal profile for raw format) • 0x06 – Custom Decoded mode (allows a customizable terminal profile for decoded format) Other values are reserved for future use.
Type	0x11			1	Custom Terminal Profile Data
Length	Var			2	
Value	→	uint8	custom_tp_len	1	Number of sets of the following elements: • custom_tp
		opaque	custom_tp	Var	Custom terminal profile, encoded as in [S1], Section 5.2. The first byte of the TP bitmask starts from custom_tp[0]. This TLV is used only for custom modes and ignored in all other cases

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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

3.19.3 Description of QMI_CAT_GET_CONFIGURATION REQ/RESP

This message is used to get the configuration mode of the QMI_CAT service currently in use. If a new configuration mode is being set by QMI_CAT_SET_CONFIGURATION_REQ, the setting might only take effect after a reboot.

3.20 QMI_CAT_GET_CACHED_PROACTIVE_CMD

Retrieves a cached proactive command from the modem.

CAT message ID

0x002F

Version introduced

Major - 2, Minor - 26

3.20.1 Request - QMI_CAT_GET_CACHED_PROACTIVE_CMD_REQ

Message type

Request

Sender

Control Point

Mandatory TLVs

Name	Version introduced	Version last modified
ID of the Proactive Command	2.26	2.26

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x01			1	ID of the Proactive Command
Length	4			2	
Value	→	enum	command_id	4	ID of the proactive command: • CAT_CACHED_COMMAND_ID_ SETUP_MENU (0x01) – Setup menu • CAT_CACHED_COMMAND_ID_ SETUP_EVENT_LIST (0x02) – Setup event list • CAT_CACHED_COMMAND_ID_ SETUP_IDLE_TEXT (0x03) – Setup Idle mode text Other values are reserved for future use.

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.26	2.26

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Slot
Length	1			2	
Value	→	enum8	slot	1	Indicates the slot to be used: • 0x01 – Slot 1 • 0x02 – Slot 2 • 0x03 – Slot 3 • 0x04 – Slot 4 • 0x05 – Slot 5 Other values are reserved for future use.

3.20.2 Response - QMI_CAT_GET_CACHED_PROACTIVE_CMD_RESP

Message type

Response

Sender

Service

Mandatory TLVs

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

Name	Version introduced	Version last modified
Result Code	2.26	2.26

Optional TLVs

The following TLVs are optional. The TLV is present if the requested proactive command is available

Name	Version introduced	Version last modified
Setup Menu Event	2.26	2.26
Setup Event List Raw Event	2.26	2.26
Setup Idle Mode Text Event	2.26	2.26

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x10			1	Setup Menu Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_menu
		opaque	pc_setup_menu	Var	Setup Menu proactive command, encoded as in [S1], Section 6.6.7.

Field	Field value	Field type	Parameter	Size (byte)	Description
Type	0x11			1	Setup Event List Raw Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_event_list
		opaque	pc_setup_event_list	Var	Setup Event List proactive command, encoded as in [S1], Section 6.6.16.
Type	0x12			1	Setup Idle Mode Text Event
Length	Var			2	
Value	→	uint32	uim_ref_id	4	Proactive command reference ID.
		uint16	cmd_len	2	Number of sets of the following elements: • pc_setup_idle_mode_text
		opaque	pc_setup_idle_mode_text	Var	Setup Idle mode text proactive command, encoded as in [S1], Section 6.6.22.

Error codes

QMI_ERR_NONE	No error in the request
QMI_ERR_INTERNAL	An 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	One of the parameters specified contains an invalid value
QMI_ERR_MISSING_ARG	One or more required TLVs are missing
QMI_ERR_INFO_UNAVAILABLE	Information is not available
QMI_ERR_OP_DEVICE_ UNSUPPORTED	Device does not support the operation

3.20.3 Description of QMI_CAT_GET_CACHED_PROACTIVE_CMD REQ/RESP

This message retrieves the requested cached proactive command from the modem.

The cached proactive command data is returned to the control point in raw format. If the QMI_CAT configuration mode in NV 65683 is neither Android mode nor Custom Raw mode, QMI_ERR_OP_DEVICE_UNSUPPORTED is returned.

A terminal response is not expected for the cached proactive command.

If the optional TLV Slot is missing, the control point assumes that the proactive command was received on slot 1.

Note: If QMI_ERR_INVALID_OPERATION is returned for the MT Call, Call Connected, or Call Disconnected ENVELOPES, it means the requested ENVELOPE is not part of the SETUP_EVENT_LIST. The operation requested by the client was not carried out and the error is not due to the envelope request packet sent by the control point. See Section 2.4.6 for more details.

A QMI_CAT Work Flow

Figures A-1 through A-10 illustrate the QMI_CAT work flow.

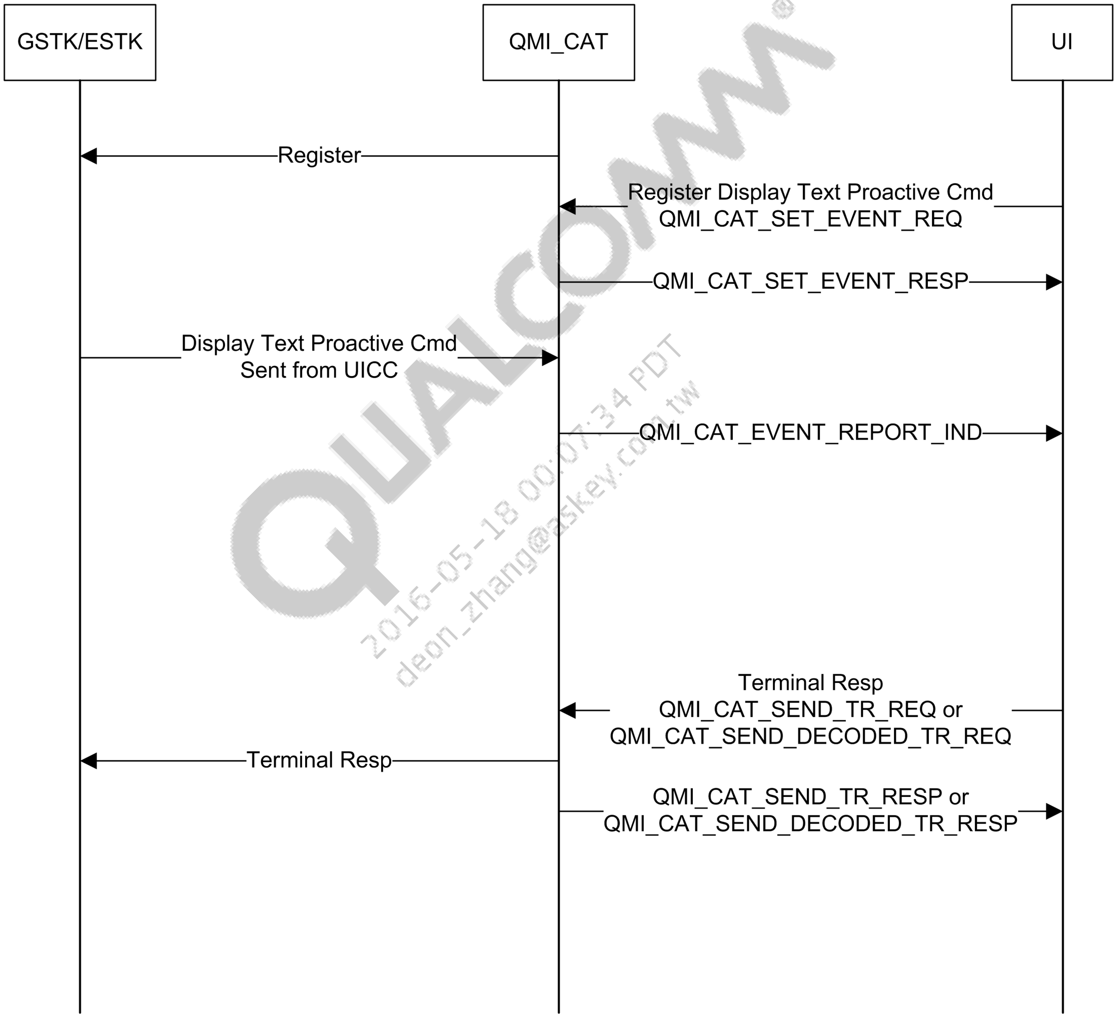

Figure A-1 Display text

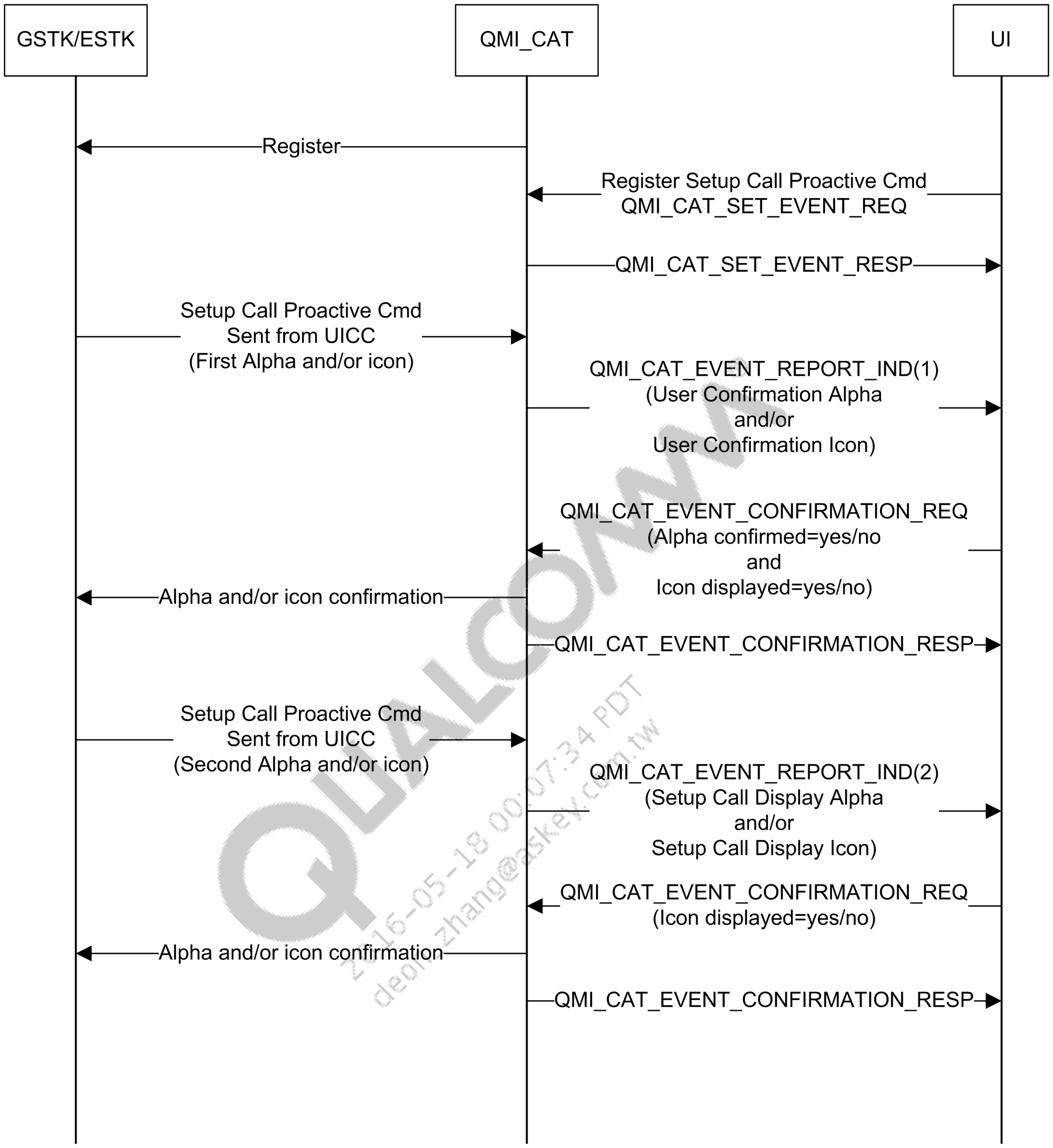
GSTK/ESTK
QMI_CAT
UI
Register
Register Setup Call Proactive Cmd
QMI_CAT_SET_EVENT_REQ
QMI_CAT_SET_EVENT_RESP
Setup Call Proactive Cmd
Sent from UICC
(First Alpha and/or icon)
QMI_CAT_EVENT_REPORT_IND(1)
(User Confirmation Alpha
and/or
User Confirmation Icon)
QMI_CAT_EVENT_CONFIRMATION_REQ
(Alpha confirmed=yes/no
and
Icon displayed=yes/no)
Alpha and/or icon confirmation
QMI_CAT_EVENT_CONFIRMATION_RESP
Setup Call Proactive Cmd
Sent from UICC
(Second Alpha and/or icon)
QMI_CAT_EVENT_REPORT_IND(2)
(Setup Call Display Alpha
and/or
Setup Call Display Icon)
QMI_CAT_EVENT_CONFIRMATION_REQ
(Icon displayed=yes/no)
Alpha and/or icon confirmation
QMI_CAT_EVENT_CONFIRMATION_RESP

Figure A-2 Set up call with two alpha identifiers

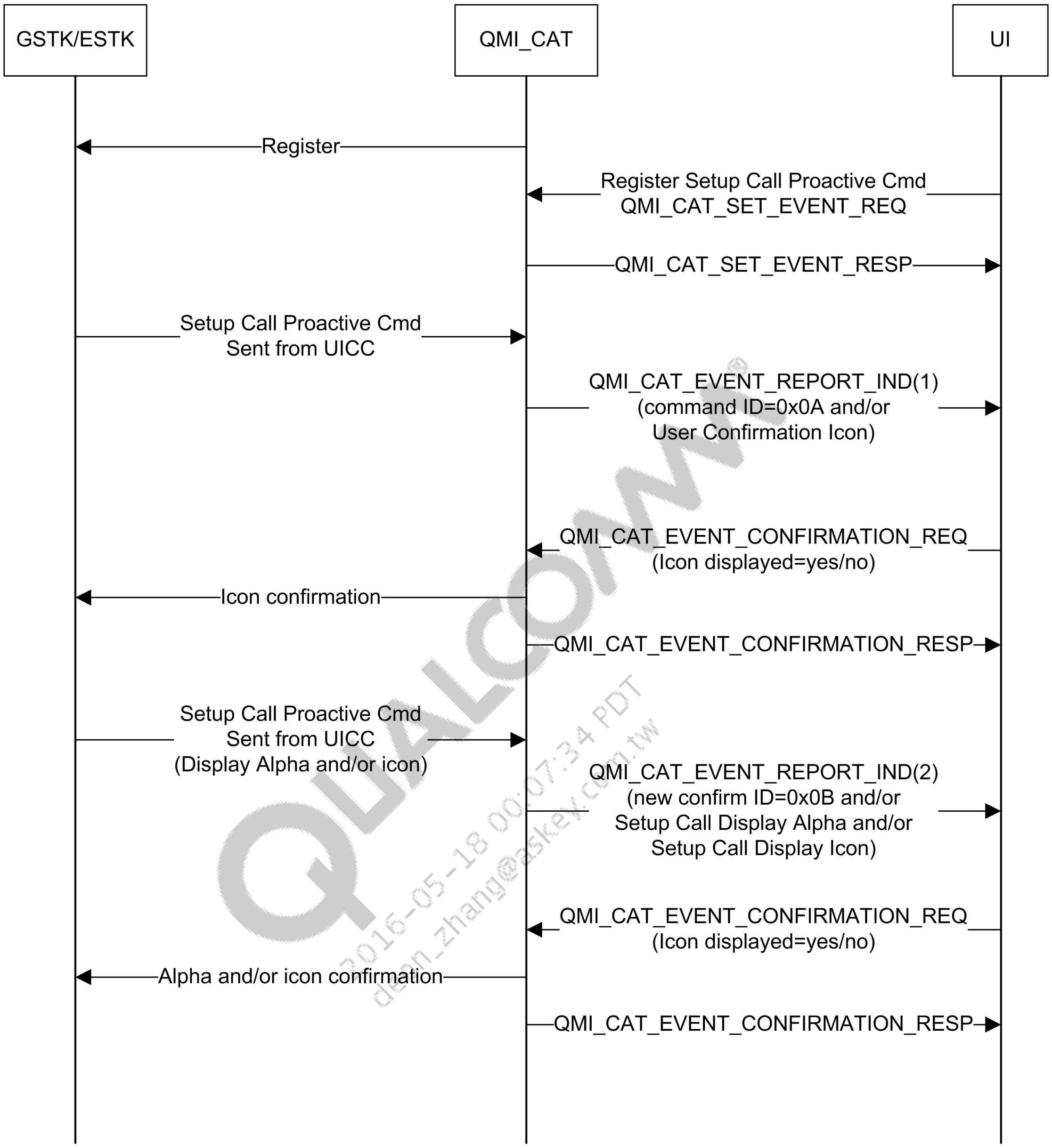
GSTK/ESTK
QMI_CAT
UI
Register
Register Setup Call Proactive Cmd
QMI_CAT_SET_EVENT_REQ
QMI_CAT_SET_EVENT_RESP
Setup Call Proactive Cmd
Sent from UICC
QMI_CAT_EVENT_REPORT_IND(1)
(command ID=0x0A and/or
User Confirmation Icon)
QMI_CAT_EVENT_CONFIRMATION_REQ
(Icon displayed=yes/no)
Icon confirmation
QMI_CAT_EVENT_CONFIRMATION_RESP
Setup Call Proactive Cmd
Sent from UICC
(Display Alpha and/or icon)
QMI_CAT_EVENT_REPORT_IND(2)
(new confirm ID=0x0B and/or
Setup Call Display Alpha and/or
Setup Call Display Icon)
QMI_CAT_EVENT_CONFIRMATION_REQ
(Icon displayed=yes/no)
Alpha and/or icon confirmation
QMI_CAT_EVENT_CONFIRMATION_RESP

Figure A-3 Set up call with display alpha identifier only

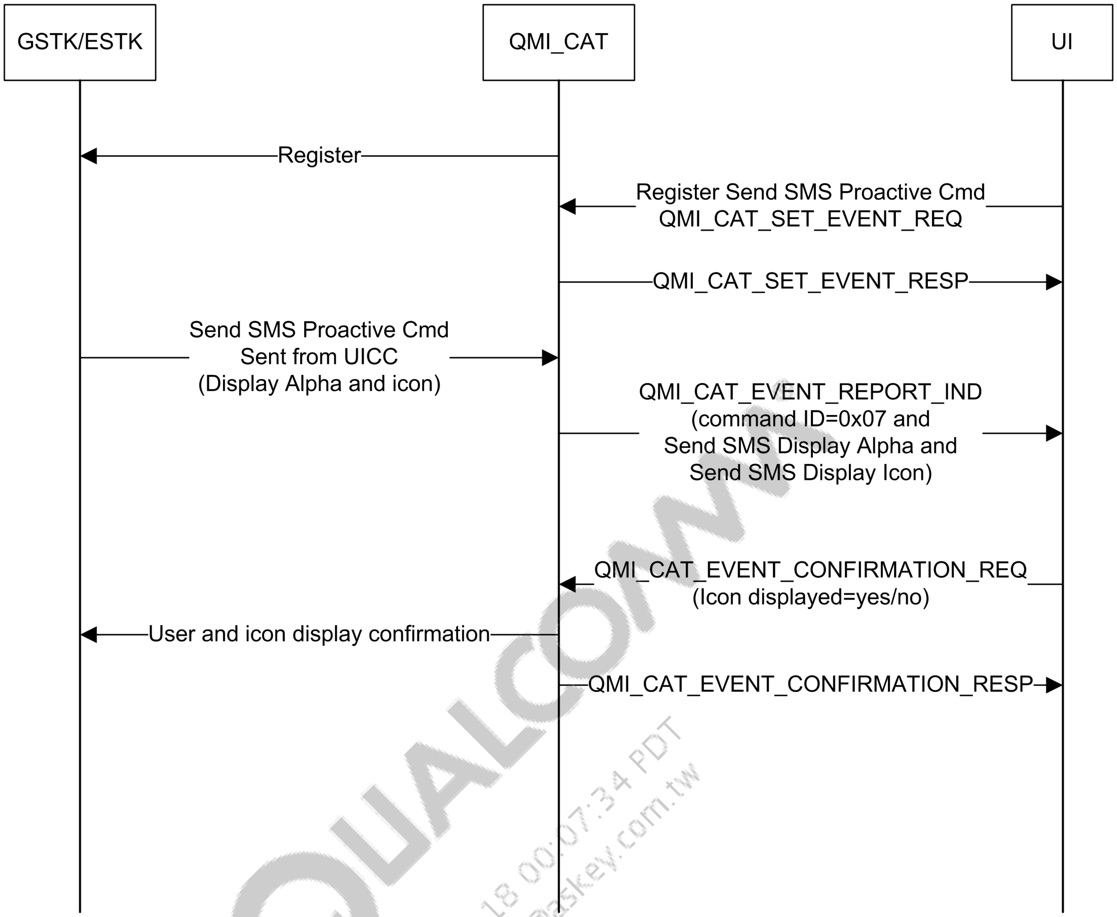

Figure A-4 Send SMS with display alpha identifier and display icon

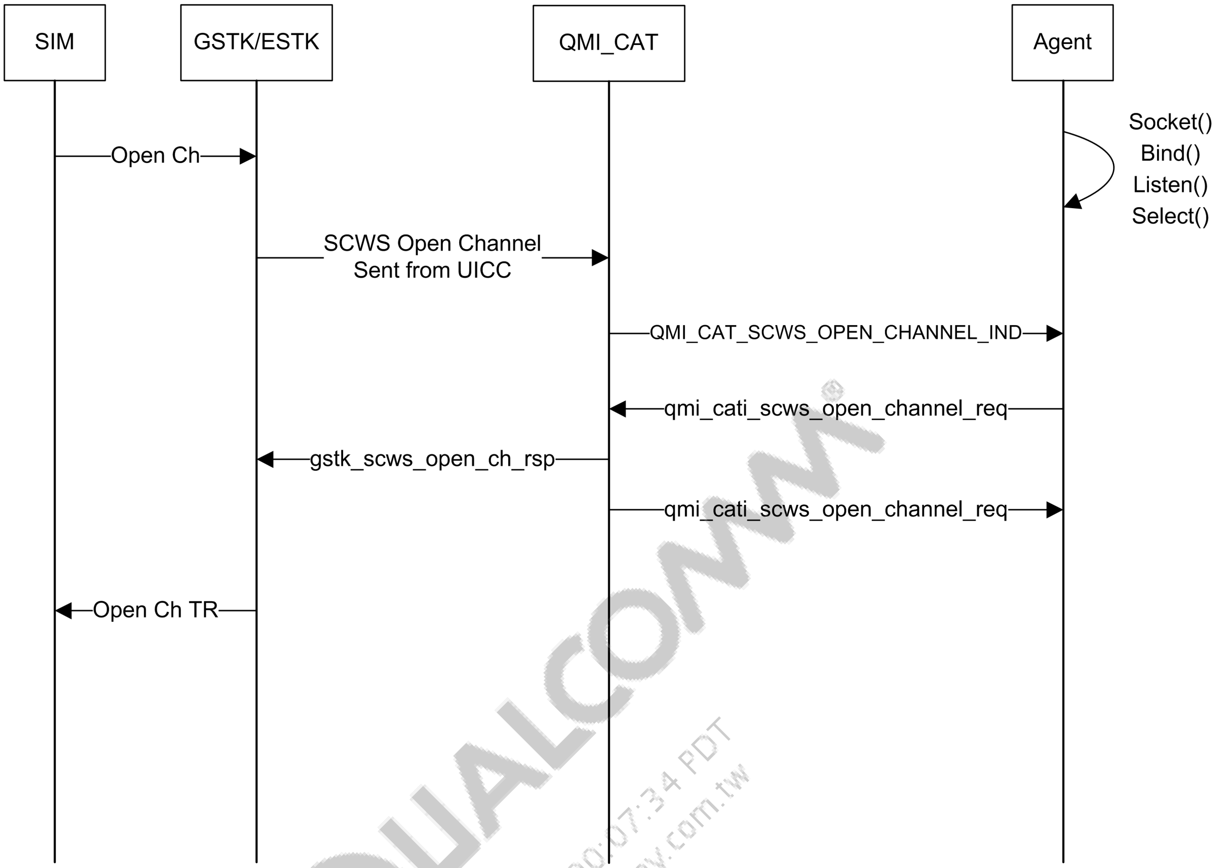

Figure A-5 SCWS open channel

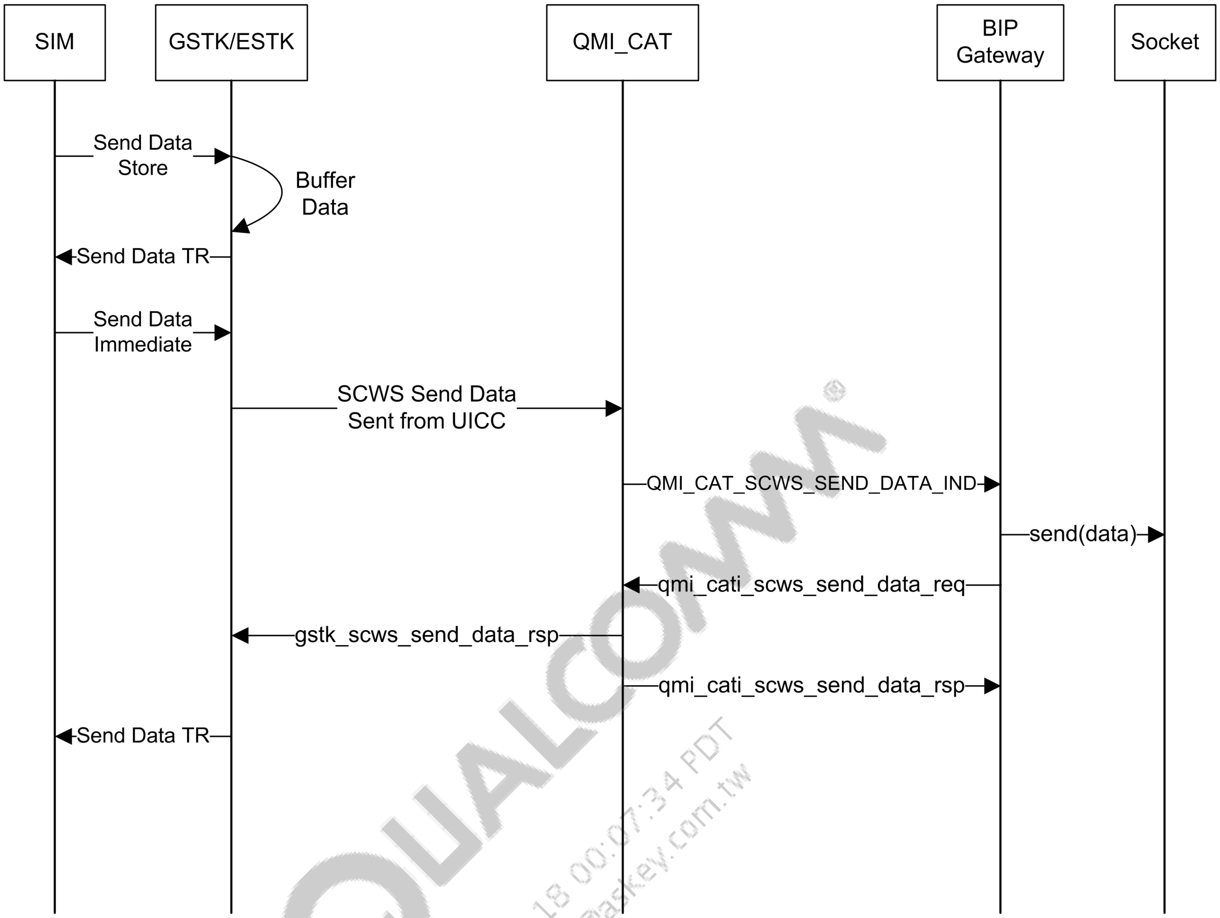

Figure A-6 SCWS send data

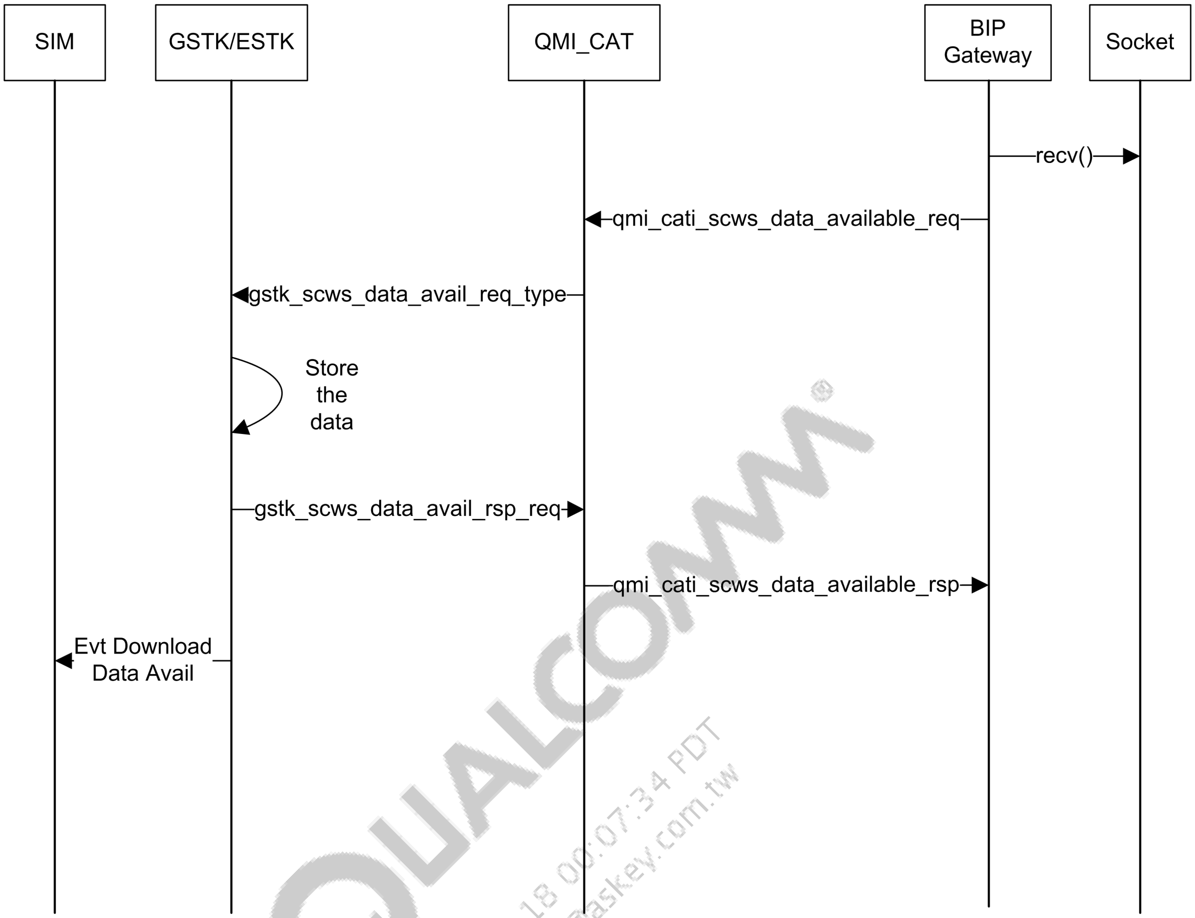
SIM
GSTK/ESTK
QMI_CAT
BIP Gateway
Socket
recv()
qmi_cati_scws_data_available_req
gstk_scws_data_avail_req_type
Store the data
gstk_scws_data_avail_rsp_req
qmi_cati_scws_data_available_rsp
Evt Download Data Avail

Figure A-7 SCWS data available

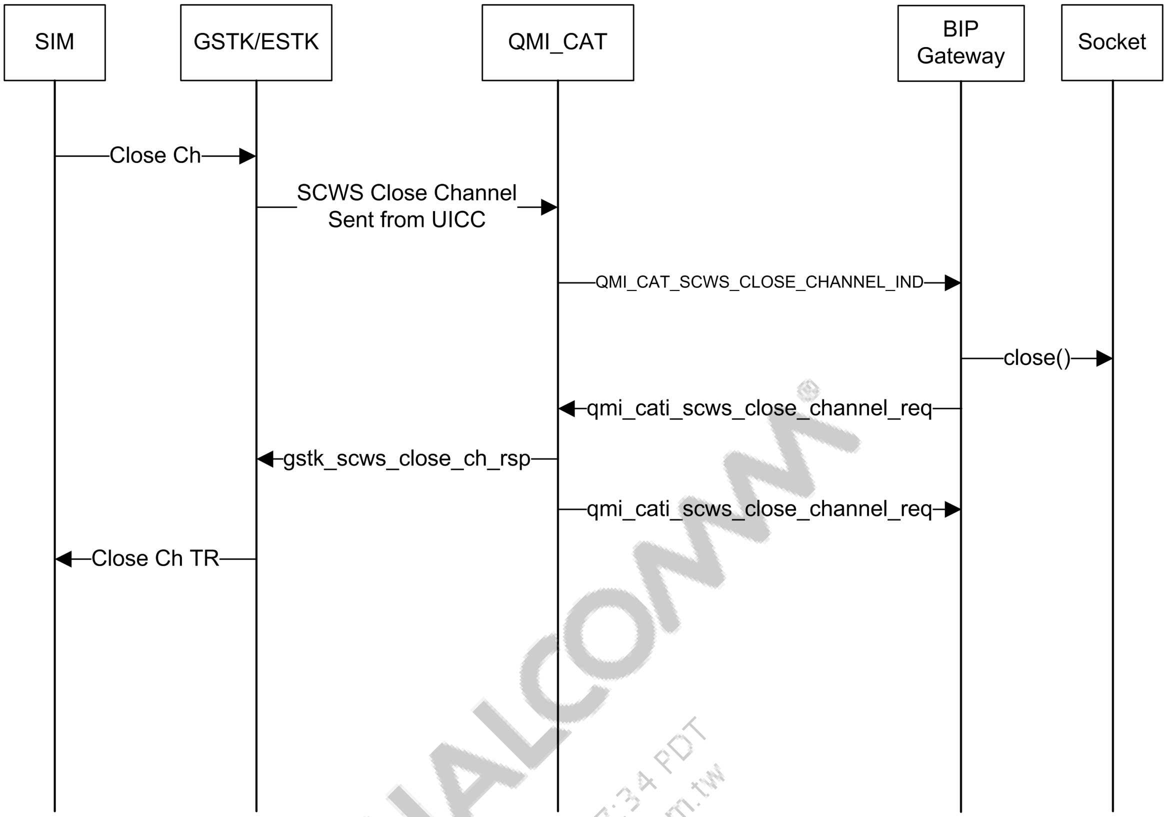
SIM
GSTK/ESTK
QMI_CAT
BIP Gateway
Socket
Close Ch
SCWS Close Channel Sent from UICC
QMI_CAT_SCWS_CLOSE_CHANNEL_IND
close()
qmi_cati_scws_close_channel_req
gstk_scws_close_ch_rsp
qmi_cati_scws_close_channel_req
Close Ch TR

Figure A-8 SCWS close channel

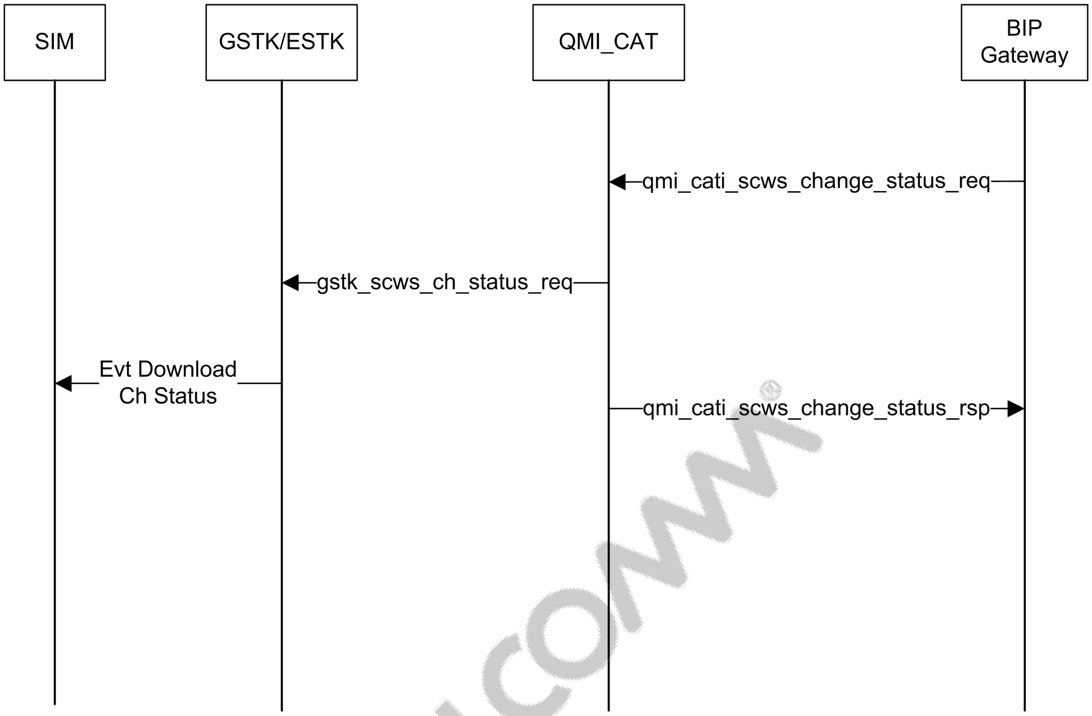

Figure A-9 SCWS channel status

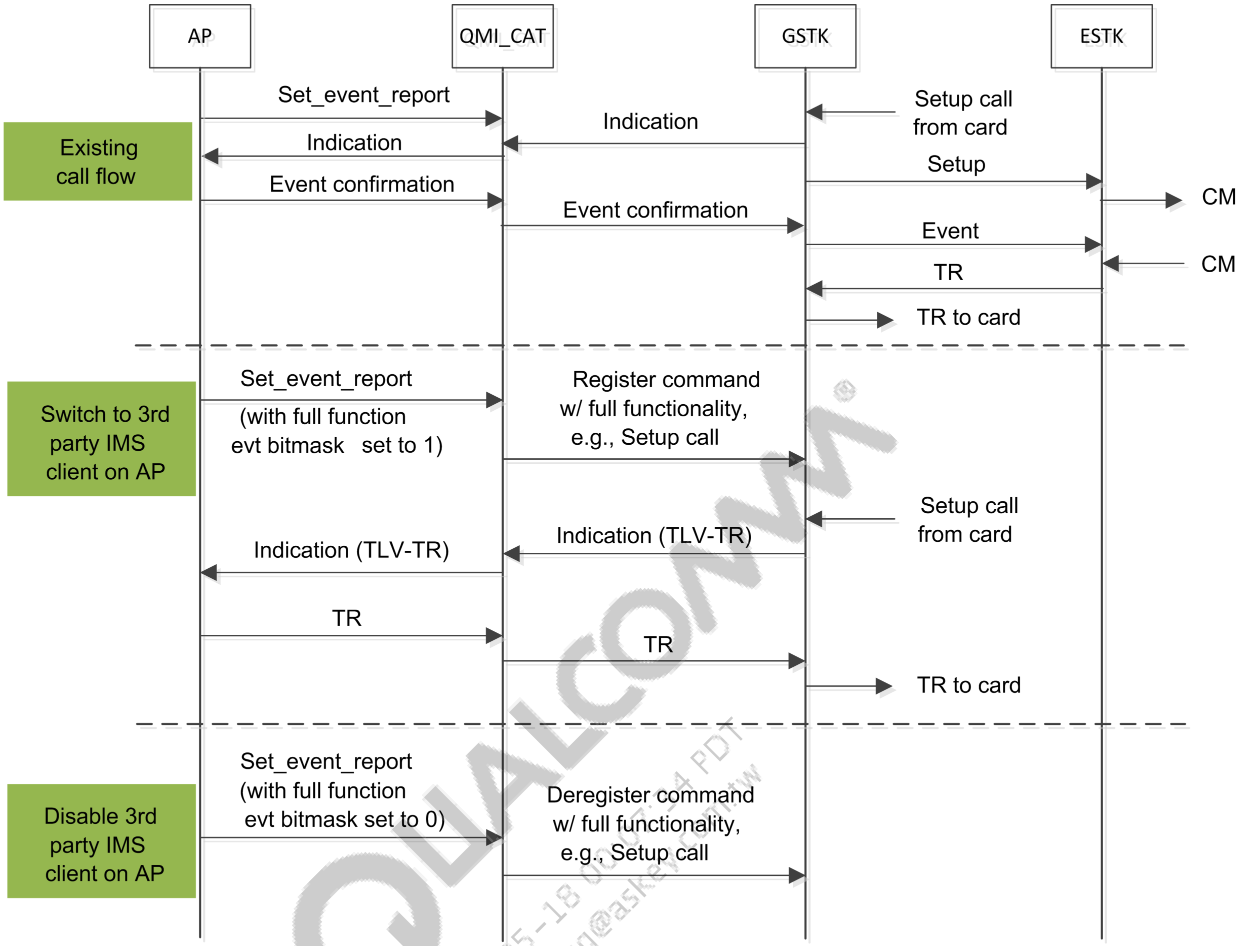
AP
QMI_CAT
GSTK
ESTK
Existing call flow
Set_event_report
Indication
Event confirmation
Indication
Event confirmation
Setup call from card
Setup
CM
Event
TR
CM
TR to card
Switch to 3rd party IMS client on AP
Set_event_report
(with full function evt bitmask set to 1)
Register command w/ full functionality, e.g., Setup call
Setup call from card
Indication (TLV-TR)
Indication (TLV-TR)
TR
TR
TR to card
Disable 3rd party IMS client on AP
Set_event_report (with full function evt bitmask set to 0)
Deregister command w/ full functionality, e.g., Setup call

Figure A-10 Routing full-function events for third party IMS clients on the AP

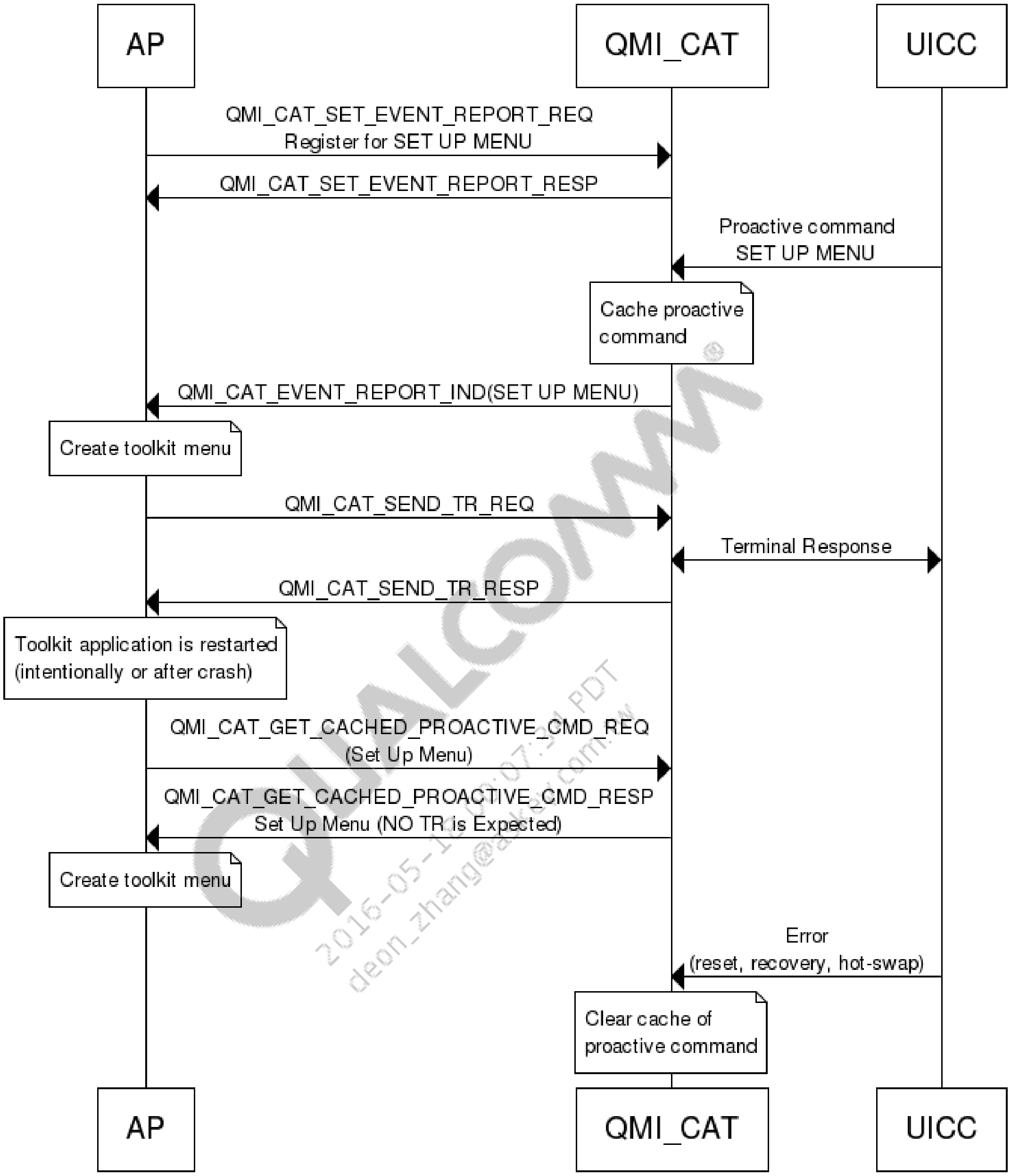
AP
QMI_CAT
UICC
QMI_CAT_SET_EVENT_REPORT_REQ
Register for SET UP MENU
QMI_CAT_SET_EVENT_REPORT_RESP
Proactive command
SET UP MENU
Cache proactive command
QMI_CAT_EVENT_REPORT_IND(SET UP MENU)
Create toolkit menu
QMI_CAT_SEND_TR_REQ
Terminal Response
QMI_CAT_SEND_TR_RESP
Toolkit application is restarted (intentionally or after crash)
QMI_CAT_GET_CACHED_PROACTIVE_CMD_REQ
(Set Up Menu)
QMI_CAT_GET_CACHED_PROACTIVE_CMD_RESP
Set Up Menu (NO TR is Expected)
Create toolkit menu
Error
(reset, recovery, hot-swap)
Clear cache of proactive command
AP
QMI_CAT
UICC

Figure A-11 Flow for Get Cached Proactive Command

B Supplementary TLVs

B.1 Display Text Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Text String	2.0	2.0
High Priority	2.0	2.0
User Control	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Icon	2.0	2.0
Immediate Response Required	2.0	2.0
Duration	2.0	2.0
Slot	2.2	2.20

B.2 Get Inkey Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Text String	2.0	2.0
Response Format	2.0	2.0
Help Available	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Icon	2.0	2.0
Duration	2.0	2.0
Slot	2.2	2.20

B.3 Get Input Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Text String	2.0	2.0
Response Format	2.0	2.0
Response Packing Format	2.0	2.0
Response Length	2.0	2.0
Help Available	2.0	2.0
Show User Input	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Default Text	2.0	2.0
Icon	2.0	2.0
Slot	2.2	2.20

B.4 Play Tone Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Tone	2.0	2.9
Duration	2.0	2.0
Icon	2.0	2.0
Slot	2.2	2.20

B.5 Setup Menu Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Help Available	2.0	2.0
Softkey Selection	2.0	2.0
Items	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Default Item	2.0	2.0
Next Action Indicator	2.0	2.0
Icon	2.0	2.0
Icon ID List	2.0	2.12
Slot	2.0	2.20
Items with DCS	2.8	2.8

B.6 Select Item Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Help Available	2.0	2.0
Presentation	2.0	2.0
Softkey Selection	2.0	2.0
Items	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Default Item	2.0	2.0
Next Action Indicator	2.0	2.0
Icon	2.0	2.0
Icon ID List	2.0	2.12
Slot	2.2	2.20
Items with DCS	2.8	2.8

B.7 Send Short Message Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Packing Required	2.0	2.0
SMS TPDU	2.0	2.0
Is CDMA SMS	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Address	2.5	2.5
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20
Response Type	2.18	2.18

B.8 Setup Call Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Call Setup Requirement	2.0	2.0
Address	2.5	2.5
Redial	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
User Confirmation Alpha	2.0	2.0
Setup Call Display Alpha	2.0	2.0
User Confirmation Icon	2.0	2.0
Setup Call Display Icon	2.0	2.0
Subaddress	2.5	2.5
Capability Configuration	2.0	2.0
Slot	2.2	2.20
Response Type	2.18	2.18

B.9 Setup Idle Mode Text Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Text String	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Icon	2.0	2.0
Slot	2.2	2.20

B.10 Send DTMF Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
DTMF	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20
Response Type	2.18	2.18

B.11 Language Notification Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Specific Language Notification	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Language	2.0	2.0
Slot	2.2	2.20

B.12 Launch Browser Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Launch Mode	2.0	2.0
URL	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Browser ID	2.0	2.0
Bearer List	2.0	2.0
Provisioning Files	2.0	2.0
Gateway Proxy	2.0	2.0
User Confirmation Alpha	2.0	2.0
Icon	2.0	2.0
Slot	2.2	2.20

B.13 Send SS Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Address	2.5	2.5

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20
Response Type	2.18	2.18

B.14 Send USSD Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
USSD String	2.5	2.5

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20
Response Type	2.18	2.18

B.15 Setup Event List Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Notification Required	2.0	2.0

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

B.16 Open Channel Decoded

B.16.1 Open Channel Related to Packet Data Service Bearer

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
On Demand Link Establish	2.4	2.4
Bearer Description (CDS/GPRS/EUTRAN External Parameter/Mapped UTRAN PS)	2.4	2.4
Buffer Size	2.4	2.4

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Network Access Name	2.4	2.4
Other Address	2.4	2.4
User Login	2.4	2.4
User Password	2.4	2.4
Transport Level	2.4	2.4
Data Destination Address	2.4	2.4
Slot	2.2	2.20

B.16.2 Open Channel Related to Default (Network) Bearer

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
On Demand Link Establish	2.4	2.4
Bearer Description (CDS/GPRS/EUTRAN External Parameter/Mapped UTRAN PS)	2.4	2.4
Buffer Size	2.4	2.4

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Other Address	2.4	2.4
User Login	2.4	2.4
User Password	2.4	2.4
Transport Level	2.4	2.4
Data Destination Address	2.4	2.4
Slot	2.2	2.20

B.17 Close Channel Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Channel ID	2.4	2.4

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20

B.18 Receive Data Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Channel ID	2.4	2.4
Channel Data Length	2.4	2.4

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20

B.19 Send Data Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Send Data Immediately	2.4	2.4
Channel ID	2.4	2.4
Channel Data	2.4	2.4

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20

B.20 Provide Local Info - Language

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

B.21 Activate

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23
Activate Descriptor Target	2.9	2.9

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

B.22 Bearer Independent Protocol Status Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Bearer Independent Protocol Status	2.22	2.22

Optional TLVs

Name	Version introduced	Version last modified
Slot	2.2	2.20

B.23 Refresh Decoded

TLVs that must always be present

Name	Version introduced	Version last modified
Decoded Header ID	2.0	2.23

Optional TLVs

Name	Version introduced	Version last modified
Alpha	2.5	2.5
Icon	2.0	2.0
Slot	2.2	2.20

C Table of Application Responses

Table C-1 lists the application responses when a proactive command is received from the card.

Table C-1 Application responses

Proactive command	Application response
Display Text	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Get Inkey	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Get Input	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Setup Menu	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Select Item	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Send SMS	QMI_CAT_EVENT_CONFIRMATION
Setup Event – User Activity	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Setup Event – Idle Screen Notify	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Setup Event – Language Select Notify	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Setup Idle Mode Text	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Setup Event List	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Language Notification	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Play Tone	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Setup Call	QMI_CAT_EVENT_CONFIRMATION
Launch Browser	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Send SS	QMI_CAT_EVENT_CONFIRMATION
Send USSD	QMI_CAT_EVENT_CONFIRMATION
Send DTMF	QMI_CAT_EVENT_CONFIRMATION
Open Channel	QMI_CAT_EVENT_CONFIRMATION
Close Channel	QMI_CAT_EVENT_CONFIRMATION
Receive Data	QMI_CAT_EVENT_CONFIRMATION
Send Data	QMI_CAT_EVENT_CONFIRMATION

Table C-1 Application responses (cont.)

Proactive command	Application response
Provide Local Info – Language	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Activate	QMI_CAT_SEND_TR or QMI_CAT_SEND_DECODED_TR
Refresh Alpha	QMI_CAT_EVENT_CONFIRMATION

D Envelope Command TLVs

The following tables list the mandatory and optional TLVs for envelope commands applicable to QMI_CAT_SEND_DECODED_ENVELOPE_CMD_REQ.

TLVs for Menu Selection

Name	Status
Envelope Command	Mandatory
Item Identifier	Mandatory
Help Request	Optional
Slot	Optional

TLVs for Event DL User Activity

Name	Status
Envelope Command	Mandatory
Slot	Optional

TLVs for Event DL Idle Screen Available

Name	Status
Envelope Command	Mandatory
Slot	Optional

TLVs for Event DL Language Selection

Name	Status
Envelope Command	Mandatory
Language	Mandatory
Slot	Optional

TLVs for Event DL HCI Connectivity

Name	Status
Envelope Command	Mandatory
Slot	Optional

TLVs for Call Control – Voice

Name	Status
Envelope Command	Mandatory
Address	Mandatory
Radio Access Technology	Mandatory
Subaddress	Conditional
Capability Configuration Parameter 1	Conditional
Capability Configuration Parameter 2	Conditional
Call Type	Conditional
Slot	Optional

Note: If the Call Type TLV is not present, the Subaddress, Capability Configuration Parameter 1, and Capability Configuration Parameter 2 TLVs become Mandatory. The client can set TLV len = 0 if TLV data is not available for the call control request.

TLVs for Call Control Envelope Response – Voice

Name	Status
Result Code	Mandatory
Call Control Result	Optional
Address	Optional
Subaddress	Optional
Capability Configuration Parameter 1	Optional
Capability Configuration Parameter 2	Optional
Alpha	Optional
BC Repeat Indicator	Optional

TLVs for Call Control – SS

Name	Status
Envelope Command	Mandatory
Address	Mandatory
Radio Access Technology	Mandatory
Call Type	Optional
Slot	Optional

TLVs for Call Control Envelope Response – SS

Name	Status
Result Code	Mandatory
Call Control Result	Optional
Address	Optional
Alpha	Optional
BC Repeat Indicator	Optional

TLVs for Call Control – USSD

Name	Status
Envelope Command	Mandatory
USSD String	Mandatory
Radio Access Technology	Mandatory
Call Type	Optional
Slot	Optional

TLVs for Call Control Envelope Response – USSD

Name	Status
Result Code	Mandatory
Call Control Result	Optional
USSD String	Optional
Alpha	Optional
BC Repeat Indicator	Optional

TLVs for Call Control – PDP Context Activation

Name	Status
Envelope Command	Mandatory
PDP Context Activation	Mandatory
Radio Access Technology	Mandatory
Slot	Optional

TLVs for Call Control Envelope Response – PDP Context Activation

Name	Status
Result Code	Mandatory
Call Control Result	Optional
PDP Context Activation	Optional
Alpha	Optional
BC Repeat Indicator	Optional

TLVs for Call Control – EPS PDN Connect Activation

Name	Status
Envelope Command	Mandatory
EPS PDN Connect Activation	Mandatory
Radio Access Technology	Mandatory
Slot	Optional

TLVs for Call Control Envelope Response – EPS PDN Connect Activation

Name	Status
Result Code	Mandatory
Call Control Result	Optional
EPS PDN Connect Activation	Optional
Alpha	Optional
BC Repeat Indicator	Optional

TLVs for Call Control – SMS

Name	Status
Envelope Command	Mandatory
Call Type	Mandatory
RP Address	Mandatory
TP Address	Mandatory
Radio Access Technology	Mandatory
Slot	Optional

TLVs for Call Control Envelope Response – SMS

Name	Status
Result Code	Mandatory
Call Control Result	Optional
Alpha	Optional
RP Address	Optional
TP Address	Optional

TLVs for Event DL Browser Termination

Name	Status
Envelope Command	Mandatory
Browser Termination Cause	Mandatory
Slot	Optional

TLVs for SMS-PP Download

Name	Status
Envelope Command	Mandatory
Address	Mandatory
SMS TPDU	Mandatory
Is CDMA SMS	Optional
Slot	Optional

TLVs for SMS-PP Download Envelope Response

Name	Status
Result Code	Mandatory
SMS-PP Data Download UICC Acknowledgment	Optional

TLVs for Event DL MT Call

Name	Status
Envelope Command	Mandatory
Transaction ID	Optional
Address	Conditional (refer to [S1] Section 7.5.1)
Subaddress	Conditional (refer to [S1] Section 7.5.1)
Slot	Optional

TLVs for Event DL Call Connected (MT/MO)

Name	Status
Envelope Command	Mandatory
Transaction ID	Mandatory
Slot	Optional

TLVs for Event DL Call Disconnected (Near End/Far End)

Name	Status
Envelope Command	Mandatory
Transaction ID	Mandatory
Cause	Optional
Slot	Optional