

Nokia Siemens Networks Short Message Service Center, Rel. 11.0

CIMD Interface Specification

DN03305755

Issue 1-0

Approval Date 2012-04-13



Nokia Siemens Networks is continually striving to reduce the adverse environmental effects of its products and services. We would like to encourage you as our customers and users to join us in working towards a cleaner, safer environment. Please recycle product packaging and follow the recommendations for power use and proper disposal of our products and their components.

If you should have questions regarding our Environmental Policy or any of the environmental services we offer, please contact us at Nokia Siemens Networks for any additional information.

The information in this document is subject to change without notice and describes only the product defined in the introduction of this documentation. This documentation is intended for the use of Nokia Siemens Networks customers only for the purposes of the agreement under which the document is submitted, and no part of it may be used, reproduced, modified or transmitted in any form or means without the prior written permission of Nokia Siemens Networks. The documentation has been prepared to be used by professional and properly trained personnel, and the customer assumes full responsibility when using it. Nokia Siemens Networks welcomes customer comments as part of the process of continuous development and improvement of the documentation.

The information or statements given in this documentation concerning the suitability, capacity, or performance of the mentioned hardware or software products are given "as is" and all liability arising in connection with such hardware or software products shall be defined conclusively and finally in a separate agreement between Nokia Siemens Networks and the customer. However, Nokia Siemens Networks has made all reasonable efforts to ensure that the instructions contained in the document are adequate and free of material errors and omissions. Nokia Siemens Networks will, if deemed necessary by Nokia Siemens Networks, explain issues which may not be covered by the document.

Nokia Siemens Networks will correct errors in this documentation as soon as possible. IN NO EVENT WILL Nokia Siemens Networks BE LIABLE FOR ERRORS IN THIS DOCUMENTATION OR FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL OR ANY LOSSES, SUCH AS BUT NOT LIMITED TO LOSS OF PROFIT, REVENUE, BUSINESS INTERRUPTION, BUSINESS OPPORTUNITY OR DATA, THAT MAY ARISE FROM THE USE OF THIS DOCUMENT OR THE INFORMATION IN IT.

This documentation and the product it describes are considered protected by copyrights and other intellectual property rights according to the applicable laws.

The wave logo is a trademark of Nokia Siemens Networks Oy. Nokia is a registered trademark of Nokia Corporation. Siemens is a registered trademark of Siemens AG.

Other product names mentioned in this document may be trademarks of their respective owners, and they are mentioned for identification purposes only.

Copyright © Nokia Siemens Networks 2013/3/20. All rights reserved



Important Notice on Product Safety

This product may present safety risks due to laser, electricity, heat, and other sources of danger.

Only trained and qualified personnel may install, operate, maintain or otherwise handle this product and only after having carefully read the safety information applicable to this product.

The safety information is provided in the Safety Information section in the "Legal, Safety and Environmental Information" part of this document or documentation set.

The same text in German:



Wichtiger Hinweis zur Produktsicherheit

Von diesem Produkt können Gefahren durch Laser, Elektrizität, Hitzeentwicklung oder andere Gefahrenquellen ausgehen.

Installation, Betrieb, Wartung und sonstige Handhabung des Produktes darf nur durch geschultes und qualifiziertes Personal unter Beachtung der anwendbaren Sicherheitsanforderungen erfolgen.

Die Sicherheitsanforderungen finden Sie unter „Sicherheitshinweise“ im Teil „Legal, Safety and Environmental Information“ dieses Dokuments oder dieses Dokumentationsatzes.

Table of Contents

This document has 59 pages.

1	About this document	7
1.1	Technical support	7
1.2	Summary of changes	7
1.3	Document accessibility	7
1.4	Typographical conventions used in documents	7
1.5	References	9
2	CIMD2 interface overview	10
3	CIMD operations	11
3.1	Application types	11
3.2	Operations from application to the SMS Center	11
3.3	Operations from the SMS Center to application	12
3.4	Operations supported by the SMS Center and different types of applications	12
3.5	Operation codes	13
4	CIMD protocol messages	15
4.1	Message format	15
4.1.1	Header	15
4.1.2	Data	16
4.1.3	Trailer	16
4.2	Message packet numbering	17
4.3	Checksum	17
4.4	Response messages	18
4.4.1	Positive response message	18
4.4.2	Negative response message	19
4.4.3	NACK message	19
4.4.4	General error response message	19
5	Parameters related to CIMD operations	20
5.1	Login (01)	20
5.2	Logout (02)	20
5.3	Submit message (03)	21
5.4	Submit status report (13)	23
5.5	Enquire message status (04)	25
5.6	Delivery request (05)	26
5.7	Cancel message (06)	28
5.8	Deliver message (20)	29
5.9	Deliver status report (23)	30
5.10	Set (08)	31
5.11	Get (09)	32
5.12	Alive (40)	32
5.13	Additional operations	33
5.13.1	General error response (98)	33
5.13.2	NACK (99)	33

6	Windowing for applications	35
6.1	Windowing scenarios	35
6.1.1	Submitting more than one short message from application	35
6.2	Windowing restrictions	36
7	CIMD parameters	37
7.1	Parameter types	37
7.2	Parameters	38
8	Error codes	44
8.1	General error codes	44
8.2	Status Error Code values	46
9	Character conversion for user data.	48
9.1	Default character conversion	48
9.2	Extension character conversion	52
9.3	Traffic direction specific character conversion	53
10	Appendix Examples	54
10.1	CIMD application use cases	54
10.2	Example sessions	58
10.2.1	Example session for a send-only application	58
10.2.2	Example session for querying application.	58
10.2.3	Example session for receiving application	59

List of Figures

Figure 1	CIMD protocol message format	15
Figure 2	Flow of delivery request operations	27

List of Tables

Table 1	Typographical conventions used in documents	8
Table 2	Operations from application to the SMS Center	11
Table 3	Operations from the SMS Center to application	12
Table 4	Operations supported by different types of applications and the SMS Center	13
Table 5	Operation codes (application request — SMS Center response)	13
Table 6	Operation codes (SMS Center request — application response)	14
Table 7	Parameter symbols	20
Table 8	Login parameters	20
Table 9	Negative response parameters for login	20
Table 10	Negative response parameters for logout	21
Table 11	Submit parameters	21
Table 12	Positive response parameters for submit	22
Table 13	Negative response parameters for submit	22
Table 14	Submit status report parameters	23
Table 15	Values for status code	24
Table 16	Positive response parameters for submit SR	24
Table 17	Negative response parameters for submit SR	24
Table 18	Enquire message status parameters	25
Table 19	Positive response parameters for enquire message status	25
Table 20	Negative response parameters for enquire message status	26
Table 21	Delivery request parameter	26
Table 22	Positive response parameters for delivery request	27
Table 23	Negative response parameters for delivery request	27
Table 24	Cancel message parameters	28
Table 25	Negative response parameters for cancel message	29
Table 26	Deliver message parameters	29
Table 27	Negative response parameter for delivery message	30
Table 28	Deliver status report parameters	31
Table 29	Set parameter	31
Table 30	Negative response parameters for set	31
Table 31	Positive response parameter for get	32
Table 32	Negative response parameters for get	32
Table 33	Operation parameters	33
Table 34	Windowing scenario 1	35
Table 35	Windowing scenario 2	36
Table 36	Parameters	38
Table 37	Parameter error codes and error texts	44
Table 38	Status Error Code parameter for SMS Center connections	46
Table 39	Default character conversion for user data	48
Table 40	Extension character conversion for user data	52

1 About this document

This document describes the CIMD2 interface, which transfers messages between applications and the Nokia Siemens Networks Short Message Service Center (SMS Center). CIMD2 is version 2 of the Computer Interface to Message Distribution protocol.

This document is intended for operator personnel involved in the daily operations of Nokia Siemens Networks SMS Center and for application developers.

1.1 Technical support

For technical support, contact local Nokia Siemens Networks.

1.2 Summary of changes

Release Date	Issue	Changes
2012-12	1-0 en	First version for Nokia Siemens Networks SMS Center 11.0.2 publishing Change the description of alphanumeric originating address in Parameters .

1.3 Document accessibility

The product documentation is delivered in PDF format, and published on Nokia Siemens Networks Online Services (NOLS) system.

To download the Nokia Siemens Networks SMS Center documents from the NOLS, perform the following steps:

1. Open the NOLS link <https://online.portal.nokiasiemensnetworks.com> in the web browser.
2. Enter NOLS Username and Password for login.



If you do not have the NOLS account, fill in the **User Registration Request** form and submit it. NOLS team will contact you.

3. Click on the **Product Information** menu. The **Product Information Center** window appears.
4. Click on the **Product finder** tab, and then in **Categories** pane, select **Service Enablers**.
5. In the **Subcategories** pane, select **Messaging**.
6. In the **Products** pane, select **Short Message Service Center**.
7. Click **Go**. The **Nokia Siemens Networks Short Message Service Center Information Center** window appears.
8. Click on the **Documentation** tab, and select the SMS Center release version to download the document set.

1.4 Typographical conventions used in documents

The following table provides the conventions used in the document:


Conventions	Description
Courier	<p>Courier is used to denote the following:</p> <ul style="list-style-type: none"> • Commands and inputs For example: <code><onstat -</code> • Paths and filenames For example: <code>/etc/opt/nokia/script</code> • Parameters and their values in configuration file descriptions For example: Define the number of rules with parameter <code>NUMBER_OF_RULES</code>. • Names of the processes For example: <code>gwrmanmx</code> • Names of the configuration files For example: <code>gwrmanmx.cf</code>
Bold	<p>Bold is used to denote the following:</p> <ul style="list-style-type: none"> • Steps in the step lists For example: Click OK. • Buttons on the graphical user interface (GUI) For example: Click the Configuration button.
Italics	<p>Italics is used to denote the following:</p> <ul style="list-style-type: none"> • Links • Document titles • Chapter titles in the PDF documents <p>For example: See <i>Guide to Documentation</i> (PDF).</p>
[section]	<p>The names of the configuration file sections are placed between square brackets [and].</p> <p>For example: Scroll down to the [KERNEL] section.</p>
<variable_data>	<p>The type of required variable data is shown between angle brackets < and >.</p> <p>For example: <process_name></p>
\	<p>Backslash at the end of a code line or command line indicates that the space is insufficient and the code or command continues on the next line.</p>
	<p>Vertical bar is used between choices, for example, in variable data <xxx yy> in configuration file arguments.</p>
	<p>A warning is used to alert the user to the dangers for protecting people. Warnings concern, for example, information about dangerous voltages or actions that can result in bodily harm or injury.</p>

Table 1 Typographical conventions used in documents




Conventions	Description
	<p>A caution is used to alert the users to the damages to the system product, losses of data, or harms to the business. Cautions concern, for example, damage caused by electrostatic discharge or actions that can cause traffic loss.</p>
	<p>A note is used to alert the reader to the important information on the proper use of the product, user expectations, error situations, and corresponding actions. Notes concern, for example, unauthorized modifications to the product that might affect Nokia Siemens Networks liability, items that are expected within a product delivery but not included, and avoidance of errors during operations.</p>
	<p>A tip is used to provide suggestions to the user.</p>

Table 1 Typographical conventions used in documents (Cont.)

1.5 References

Nokia Siemens Networks SMS Center documents:

1. Configuration Files
2. External Applications Configuration Guide

Other documents:

1. 3GPP specification 23.038: Alphabets and language-specific information, Release 6.0
2. Forum Nokia at www.forum.nokia.com

2 CIMD2 interface overview

Nokia Siemens Networks SMS Center includes the CIMD2 interface, which transfers messages between applications and the SMS Center. CIMD2 is version 2 of the Computer Interface to Message Distribution protocol. The CIMD2 interface is suitable for client applications that send and retrieve messages over TCP/IP sockets.

An application is interconnected through the CIMD2 connection to an SMS Center. The main purpose of this interconnection is to transfer messages from the applications to the mobile stations (!!!!! Variable is not defined !!!!!) and from the !!!!! Variable is not defined !!!!! to the applications. Other type of information can also be conveyed over the interconnection, for example, status reports (SRs) from the GSM/GPRS network to the applications.

The system architecture discussed in this document consists of the application and the SMS Center, and the purpose of this document is to specify the interface between them.

When a message has been submitted to the SMS Center using the CIMD2 interface, the SMS Center uses its own retry policy to attempt to deliver the message. If the delivery fails, the message is stored in the SMS Center database until a delivery attempt succeeds. If a permanent error occurs or the validity period (VP) of the message expires, the message is deleted.

The CIMD2 operations are specified in [CIMD operations](#) and [Parameters related to CIMD operations](#) in *CIMD Interface Specification* (PDF). Each CIMD2 operation carries a number of parameters with it, that is, data items that specify the subscriber, some facts about the operation itself, and so on.

The coding of information related to the operations and parameters, that is, how the SMS Center communicates with the applications, is introduced in [CIMD protocol messages](#) in *CIMD Interface Specification* (PDF).

The parameters are specified in [Parameters related to CIMD operations](#) in *CIMD Interface Specification* (PDF).



In some cases you may want to restrict the use of some parameters. You can configure these parameters in the interface profiles.

In this interface specification, the operations and the parameters specified for each operation represent the maximum amount of information that the application or the SMS Center provides. Note that in most cases it is not reasonable for the application to send all the possible parameters.

3 CIMD operations

The types of operations are the following:

- operations originated by the application
- operations originated by the SMS Center
- operations that can be originated by both the application and the SMS Center

The parameters related to each operation are specified in [Parameters related to CIMD operations](#), and the values of the parameters are specified in [CIMD parameters](#) in *CIMD Interface Specification* (PDF).

3.1 Application types

The type of the application must be specified before the application starts to operate. The type, along with other information about the application, is stored in the SMS Center. When defining the operations that the different applications can use, three basic types of applications can be distinguished:

- *Send-only applications* that can only submit messages to the SMS Center. The messages can be destined to mobile stations (!!!!!! Variable is not defined !!!!!). Status reports (SRs) of the sent messages must be explicitly requested by the application.
- *Querying applications* that do not receive anything from the SMS Center automatically, but make queries if there is something to be retrieved. The application can retrieve messages coming from !!!!!! Variable is not defined !!!!! , but only on demand. The application is typically connected to the SMS Center every now and then to submit a message and can at the same time also check if there is something to be received. An example of this kind of application is a PC application with a modem connection to the SMS Center. SRs, if demanded, must be explicitly requested by the application.
- *Receiving applications* that are always ready to receive messages if the SMS Center has messages or SRs to send to it. A receiving application can automatically receive messages stored in the SMS Center when logging in to the SMS Center. This allows fast two-way message exchange between the application and the SMS Center. The receiving applications can also be set up in such a way that when they are connected to an SMS Center, messages stored in the SMS Center are not sent automatically when the application logs in. In such cases, the application receives new incoming messages, but it has to ask for any old messages stored in the SMS Center.

3.2 Operations from application to the SMS Center

The following table lists the operations from an application to the SMS Center.

Operation	Definition
Login	Is used by all applications before any operations.

Table 2 Operations from application to the SMS Center

Operation	Definition
Logout	Is used by all applications to indicate the end of the session. The logout operation does not release the connection. Therefore, a disconnect must be explicitly performed after the logout.
Submit	Is used by the application to send messages to !!!!! Variable is not defined !!!!! and/or other applications. In the submit operation, the application can request transmission of SRs for the message.
Submit status report	Is used by the application to explicitly generate status reports to !!!!! Variable is not defined !!!!! and/or other applications.
Delivery request	Is used by the application to retrieve messages.
Cancel	Is used by the application to cancel messages that it has sent.
Enquire message status	Is used by the application to request an SR for a previously submitted message.
Set	Is used by the application to change the values of those interface parameters that it is allowed to change.
Get	Is used by the application to enquire the values of the interface parameters.
Alive	Is used by the application to check whether the link to the SMS Center is alive, and to keep the connection alive during long idle periods. The alive operation prevents automatic logout of the application.

Table 2 Operations from application to the SMS Center (Cont.)

3.3 Operations from the SMS Center to application

The following table lists the operations from the SMS Center to an application.

Operation	Definition
Deliver message	Is used by the SMS Center for delivering a message to the application automatically.
Deliver status report	Is used by the SMS Center to send an SR that describes the current status of a message sent by the application.

Table 3 Operations from the SMS Center to application

3.4 Operations supported by the SMS Center and different types of applications

The following table lists the operations supported by different types of applications and by the SMS Center.

Operation	Application type		
	Send-only	Querying	Receiving
Login	+	+	+
Logout	+	+	+
Submit	+	+	+
Submit status report	+	+	+
Enquire message status	+	+	+
Delivery request	-	+	+
Cancel	+	+	+
Deliver message	-	+	+
Deliver status report	-	-	+
Set parameters	+	+	+
Get parameters	+	+	+
Alive	+	+	+

Table 4 Operations supported by different types of applications and the SMS Center

- + supported
- not supported

3.5 Operation codes

All available operations are listed in the following tables, showing both the request and the response operation code.

Application request →	← SMS Center response
login request (01)	login response (51)
logout request (02)	logout response (52)
submit message request (03)	submit message response (53)
submit status report request (13)	submit status report response (63)
enquire message status request (04)	enquire message status response (54)
delivery request (05)	delivery request response (55)
cancel message request (06)	cancel message response (56)
set message request (08)	set response (58)
get message request (09)	get response (59)
alive request (40)	alive response (90)
	general error response (98)
	nack (99)

Table 5 Operation codes (application request — SMS Center response)

SMS Center request →	← Application response
deliver message (20)	deliver message response (70)
deliver status report (23)	deliver status report response (73)
	nack (99)

Table 6 Operation codes (SMS Center request — application response)

4 CIMD protocol messages

4.1 Message format

Each message, operation, or response consists of a header, data, and trailer part of the message, as shown in the figure below.

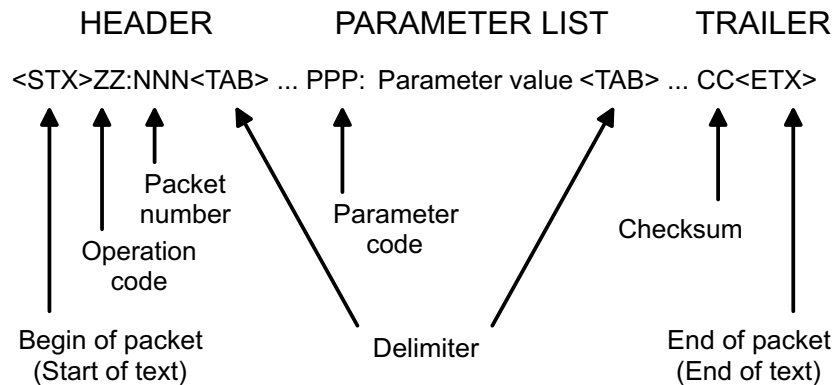


Figure 1 CIMD protocol message format



Any data transmitted between packets can be ignored. This data can originate from modems, terminal drivers, and so on.

The coding of the message parts is explained in [Header](#), [Data](#), and [Trailer](#) in *CIMD Interface Specification* (PDF).

4.1.1 Header

The header has the following format:

```
<stx>ZZ:NNN<tab>
```

where

- `<stx>` is the start-of-text indicator, which consists of a single byte that contains the decimal value 2.
- `ZZ` defines the operation code and consists of two bytes that contain the ASCII characters of the digits 0 to 9, which range from 48 to 57.
- `NNN` represents the message packet number, which consists of three bytes that contain the ASCII characters of the digits 0 to 9, which range from 48 to 57.

The `ZZ` and `NNN` fields are separated by one byte that contains the ASCII code of a colon, which is 58. The header is terminated by one byte that contains the ASCII code of `<tab>`, which is 9.

The following is an example of the decimal values for each byte of a header:

```
2 48 49 58 48 48 49 9
```

In the notation used in the rest of this document, the header looks in the following way:

```
<stx>01:001<tab>
```

4.1.2 Data

The data fields consist of a list of parameters each terminated by the <tab> character. All parameter fields have the following format:

```
PPP:value of the parm<tab>
```

where PPP indicates the parameter type and consists of three bytes that contain the ASCII values of the digits 0 to 9 (values 48 to 57). After the single byte that contains the ASCII value for the colon (58), the value of the parameter is coded with a variable number of bytes. The parameter is terminated by a single byte that contains the ASCII value for tab (9).

The coding of the parameter value depends on the type of the parameter and is explained in [CIMD parameters](#) in *CIMD Interface Specification* (PDF).

All parameters consist of the ASCII equivalents of digits or the characters of the alphabet. The parameter for the user data (033), however, can also consist of other characters, which enable the use of all characters in the default GSM character set. The reserved characters 0x00 (NUL), 0x02 (STX), 0x03 (ETX), 0x09 (TAB) are not allowed in any parameter.

The parameters allowed in a packet depend on the operation and sometimes on the user profile. The order of parameters is free, and many parameters can be omitted.

Examples of the decimal values of the bytes for a few parameters are given below, together with the notation used in this document.

Example: Password parameter (11)

```
48 49 49 58 83 101 51 114 83 116 9
011:SeCrEt<tab>
```

Example: Userdata parameter (33)

```
48 51 51 58 104 105 32 116 104 101 114
101 32 33 9
033:hi there !<tab>
```

4.1.3 Trailer

The trailer of a packet has the following format:

```
CC<etx>
```

where CC consists of two bytes that contain the checksum of the packet, and <etx> is a single byte that contains the end-of-text character, which has the value 3.

The use of the CC field is optional, in which case the trailer consists only of the single <etx> byte.

The following is an example of the decimal values of the bytes in a trailer, together with the notation used in this document.

Example: Decimal values of the bytes in a trailer

```
51 65 3
3A<etx>
```

All parts combined give the following typical message:

```
<stx>ZZ:NNN<tab>PPP:parameter1<tab>QQQ:parm2<tab><etx>
```


When real values for ZZ, NNN, and so on, are used, you receive a message of the following type:

```
<stx>01:001<tab>010:wwwstat<tab>011:wwwstat<tab><etx>
```

or:

```
<stx>03:011<tab>021:123456789<tab>033:hi there<tab><etx>
```

4.2 Message packet numbering

All CIMD2 messages are assigned a packet sequence number. This packet sequence number is used for detecting duplicate packets or missing packets. The number is assigned according to the following rules:

- Operations from the application to the SMS Center are assigned an odd packet number, starting from one. Subsequent packet numbers are incremented by two. After reaching 255, the number wraps back to one again. Using the coding described before the three-character NNN field (see the previous example), the message packet numbering for a list of application-originated packets is the following: 001, 003, 005, ... 253, 255, 001, 003, ...
- Operations from the SMS Center to the application are assigned an even packet number starting from zero. Subsequent packet numbers are incremented by two. After reaching 254, the number wraps back to zero again. This means that the message packet numbering for SMS Center-originated packets is the following: 000, 002, 004, ... 252, 254, 000, 002, 004, ...
- All response messages are assigned the same packet number as the request. Thus, the responses from the application to the SMS Center have even numbers and the responses from the SMS Center to the application have odd numbers.



Applications may not use the same packet sequence number more than once. The applications must always increment the sequence number by two for a new message.

If an application erroneously uses the same packet sequence number repeatedly for several messages, the ASE repeats the first response to all the requests that use the same sequence number.

4.3 Checksum

The checksum is an optional field, but when it is used, you can calculate it according to the following procedure:

1. At the beginning of the message, set the checksum to 0.
2. Retrieve the first byte of the message.
3. Add the value of the byte to the checksum.
4. Truncate the checksum so that it contains only the least significant byte.
5. If available, retrieve the next byte from the message and repeat step 3. The process stops when the <etx> field is found two bytes further in the message.

Example: Checksum calculation using C language

```
int GetChecksum( char *pstx,
char *petx )
/*****
/* pstx points to the in the message */
```

```

/* petx points to the  in the message          */
/*****/
{
  int checksum = 0;
  char *p = pstx;

  while (petx - p >= 2 ){
    checksum += *p;
    checksum &= 0xFF;
    p++;
  }
  return( checksum );
}

```

The previous example shows that all the characters from the first character to the last character before the checksum characters are included in the sum. Thus, <stx> is the first character in the checksum calculation, and the last <tab> before the checksum is the last character. The checksum characters and the <ETX> are not included in the calculation.

The following line indicates the characters included in the checksum calculation:

```
<stx>ZZ:NNN<TAB>PPP:value1<tab>QQQ:val2<tab>
```

The coding of the checksum value into the two bytes of the CC field is done in the following way. The most significant fourbits of the checksum are coded in the first byte, and the least significant four bits are coded into the second byte of the checksum CC field. The ASCII representation of the digits `0` to `9` and `A` to `F` are used for coding the hexadecimal value of the four bits into the message.

For example, if the checksum is 58 (decimal), which is 0x3A (hexadecimal), the most significant four bits give us the value 3, and the ASCII representation `3` has the value 51 (decimal) or 0x33 (hexadecimal). The second value gets the value `A`, which is 65.

The use of the checksum is optional for the application. The SMS Center always includes a checksum to the packets that it sends to the application.

4.4 Response messages

The interaction between an application and the SMS Center involves the sending of request messages, to which the other party responds with response messages. Depending on the situation, the response message can be one of the following:

- positive response message
- negative response message
- NACK message
- general error response message

4.4.1 Positive response message

After processing the request message, the application or the SMS Center sends back a positive response message. The operation code of the response packet is fixed to be 50 more than the operation code of the request packet. The packet number is the same as the request message.

A positive response message informs the initiator of the operation that the request message was received correctly and the operation was performed successfully.

In some exceptional cases, for example, in case of cancel message, the response message assures only that the request message was received correctly and the operation is performed, but the result of the operation is not available.

4.4.2 Negative response message

If the request contains invalid parameters or cannot be performed, a negative response message is sent back to the initiator of the operation. Positive and negative responses have the same operation code and packet number, but the negative response message has an error code and optionally an error text parameter. The usage of error texts is set in the interface profile.

A negative response message informs the initiator of the operation that the request message was received correctly, but the operation could not be performed successfully.

4.4.3 NACK message

A NACK message is a special case that performs one of the following actions:

- triggering a retransmission of the request message if the checksum calculated by the receiver does not match the checksum found in the packet
- indicating that the packet number used is not the expected number
The packet number of the NACK message always contains the expected packet number.



The SMS Center never changes the packet number because of the NACK message. The SMS Center always assumes that the packet number confusion should be corrected by the application.

4.4.4 General error response message

This response message is used if the SMS Center receives an unknown operation. It indicates the correct reception of a request with a correct checksum and packet number but an illegal operation code, for example, operation code 7.

5 Parameters related to CIMD operations

The following table lists the symbols used. For more information on the parameters, see [CIMD parameters](#) in *CIMD Interface Specification (PDF)*.

Symbol	Meaning
M	Mandatory parameter
O	Optional parameter

Table 7 Parameter symbols

5.1 Login (01)

A login operation must always be done before any other operation. The SMS Center selects the interface profile based on the user identity given in the login operation.

Number	Login parameters	Presence
010	User identity	M
011	Password	M
012	Subaddr	O
019	Window size	O
071	IP address	O

Table 8 Login parameters

Positive response for login contains no parameters.

Number	Negative response parameters
900	Error code
901	Error text (optional)

Table 9 Negative response parameters for login

Negative response can follow if the user is already logged in or the User identity or Password or Subaddr is incorrect. It can also follow if the window size exceeds the allowed limits for login.

5.2 Logout (02)

The logout operation closes the session, but not the connection. This allows an application connected through a modem to log in again without having to re-establish the modem connection to the SMS Center. In most cases, the application can close the connection without performing an explicit logout.

The logout operation does not need any parameters.

Positive response for logout contains no parameters.

Number	Negative response parameters
900	Error code
901	Error text (optional)

Table 10 Negative response parameters for logout

5.3 Submit message (03)

Submit in its simplest mode just passes the message text and destination address to the SMS Center that takes care of delivery. There are, however, also some special features that can be requested with the submit operation, for example, first delivery time, or message to many recipients.

When the application wants to submit a message, it builds the message text and places it into the parameter user data in the submit operation. The text is sent with other necessary parameters to the SMS Center. The SMS Center then sends the message to the mobile station (MS) or another application.

The submitted message can be identified afterwards by using a time stamp generated by the SMS Center (returned in a submit response) and the destination address.

Number	Submit parameters	Presence
021	Destination address (multi)	M
023	Originating address	O
027	Alphanumeric originating address	O
030	Data coding scheme	O
032	User data header (bin)	O
033	User data ¹	O
034	User data binary ¹	O
044	More messages to send ²	O
050	Validity period relative ³	O
051	Validity period absolute ³	O
052	Protocol identifier	O
053	First delivery time relative ⁴	O
054	First delivery time absolute ⁴	O
055	Reply path	O
056	Status report request	O
058	Cancel enabled	O
064	Tariff class	O
065	Service description	O
067	Priority	O

Table 11 Submit parameters

¹ the user data (033) or user data binary (034) field

- 2 the First delivery time relative or First delivery time absolute parameters affect the value -> 0
- 3 either relative or absolute validity period
- 4 either relative or absolute first delivery time

For information on how to use the user data parameters (032, 033, and 034), see [CIMD parameters](#) in *CIMD Interface Specification* (PDF).

If an originating address is given in the submit message, it is regarded as a sub-address and it is appended to the end of the address that is set in the interface profile.

Number	Positive response parameters
021	Destination address
060	Service centre time stamp (SCTS)

Table 12 Positive response parameters for submit

Number	Negative response parameters
021	Destination address *
900	Error code
901	Error text (optional)

Table 13 Negative response parameters for submit

- * The destination address(es) is included only if the SMS Center could interpret the address in the request correctly.



A submit to multiple destinations is responded to with a single submit response message. This submit response contains an element that is formed of a destination address and a time stamp (positive response) or of a destination address and an error code (negative response) for each destination address in the submit message.

Example: Submit operation to multiple destinations

```
Submit Request --->
<STX>03:007<TAB>021:11111<TAB>021:22222<TAB>021:333<TAB>033:UserData<TAB><ETX>
<-- Submit Response
<STX>53:007<TAB>021:11111<TAB>060:971107131212<TAB>
>021:22222<TAB>060:971107131212<TAB>021:333<TAB>
>900:300<TAB><ETX>
```



As the CIMD interface allows the parameters to be in any order, the decoding ends immediately after the first erroneous parameter is received. So the addresses are returned only if the error occurs after the address parameters. For more information, see the examples below.

Example: Addresses are returned even if there is an erroneous parameter

In the following case, the 056 parameter, that is, status report (SR) request, is wrongly set (too big value). Despite that, all addresses (021) are returned before an error code (900:310).

Submit:

03:005 021:123 021:12345 021:1234567890 033:hi there 056:321

Submit response:

53:005 021:123 021:12345 021:1234567890 900:310 D6

Example: Addresses are not returned because of an erroneous parameter

In the following case, the 056 parameter, that is, SR request, is wrongly set (too big value). Because the second address (021:456) comes after the erroneous parameter, it is never decoded, thus it is not added to the response at all.

Submit:

03:007 021:123 033:hello 056:333 021:456

Submit response:

53:007 021:123 900:310 20

Likewise, if the erroneous parameter happens to be before the first address, none of the addresses is returned.

5.4 Submit status report (13)

This operation is used by the application to explicitly generate status reports (SRs) when messages are redirected to other destinations but the SR needs to be related to the original destination. The operation in its simplest form needs the destination address, the originator address, the time stamp, and the message reference.

The SRs submitted by this operation cannot be queried for status.

Number	Submit status report parameters	Presence
021	Destination address	M
023	Originator address	M
050	Validity period relative for SR	O
051	Validity period absolute for SR	O
052	Protocol identifier (PID)	O
060	Service centre time stamp (SCTS)	M
061	Status code	O
062	Status error code	O
063	Discharge time	O
069	Service center address	O
070	Message reference	M

Table 14 Submit status report parameters



Status code (061) is also named delivery outcome. Its values can be any of the following ones:

Status code	Description	Result in the SMS Center kernel (Log success indicator)
0	no status available	(not supported by SubmitSR operation)
1	in process	12 (waiting for first delivery)
2	validity period expired	11 (validity period expired)
3	delivery failed	6 (permanent error)
4	delivery successful	0 (delivery OK)
5	no response	4 (temporary error)
6	last no response	6 (permanent error)
7	message canceled	9 (canceled)
8	message deleted	9 (canceled)
9	message deleted by canceling	9 (canceled)

Table 15 Values for status code



For detailed information about status error code (062), refer to Chapter 1.1 [Status Error Code values](#) of this document.

The originator address is the MSISDN that receives the SR and the destination address is the MSISDN that sends the SR. This operation differs from the Submit message operation.

Neither the destination address nor the originator address parameter can have multiple instances in the parameter list.

Number	Positive response parameters
021	Destination address
023	Originator address
060	Service centre time stamp (SCTS)

Table 16 Positive response parameters for submit SR

Number	Negative response parameters
021	Destination address
023	Originator address
900	Error code
901	Error text (optional)

Table 17 Negative response parameters for submit SR



This response (either positive or negative) is sent by ASE directly after sending the SR packet to the kernel. So it is not a guarantee of a successful delivery to the destination. It only indicates that the SMS Center has accepted it as a valid SR and tries to deliver it.

Example: Submit SR operation

```
<STX>01:001<TAB>010:test<TAB>011:test<TAB><ETX>
<STX>20:000<TAB>021:1234<TAB>023:44777030345<TAB>060:070509160949<TAB>
033:hello<TAB>052:0<TAB>030:0<TAB>028:44779213008<TAB>026:234777030000345<TAB>
069:44779242301<TAB>070:150<TAB>056:0<TAB>78<ETX>
```

Note that the MS does not request any SRs (056 = 0). Send an SR with the 013 command.

```
<STX>70:000<TAB><ETX>
<STX>13:003<TAB>021:1234<TAB>023:44777030345<TAB>060:070509160949<TAB>
070:150<TAB><ETX>
```

Note that when 021 and 023 parameters are in use, they remain the same as they were when they arrived at the Deliver message (20) operation.

```
<STX>63:003<TAB>021:1234<TAB>023:44777030345<TAB>060:070509160949<TAB>40<TAB><ETX>
<STX>02:005<TAB><ETX>
<STX>52:005<TAB>41<TAB><ETX>
```

5.5 Enquire message status (04)

This operation is used to request a status report (SR) for a previously submitted message.

An enquire message status operation can be performed independently of the status report request parameter used in the submit message operation. However, a returned status code usually indicates that the status is unknown, because the SMS Center does not keep track of statuses unless requested at submission time for performance reasons.

No multiple enquiries are allowed in one enquire message status message packet, so applications have to request each SR separately.

Number	Enquire message status parameters	Presence
021	Destination address	M
060	Service centre time stamp	M

Table 18 Enquire message status parameters

The information in the enquire message status response is the same as in the message status delivery.

Number	Positive response parameters
021	Destination address
060	Service centre time stamp
061	Status code
062	Status error code (optional)

Table 19 Positive response parameters for enquire message status

Number	Positive response parameters
063	Discharge time

Table 19 Positive response parameters for enquire message status (Cont.)

Number	Negative response parameters
900	Error code
901	Error text (optional)

Table 20 Negative response parameters for enquire message status

Example: Enquire message status

```

Enquire Message Status --->
<STX>04:003<TAB>021:DestAddr<TAB>060:ServiceCentreTimeStamp<TAB>cs<ETX>
<--- Positive response
<STX>54:003<TAB>021:DestAddr<TAB>060:ServiceCentreTimeStamp<TAB>061:StatusCode<TAB>063:DischargeTime<TAB>cs<ETX>
<--- Negative response
<STX>54:003<TAB>900:ErrorCode<TAB>checksum<ETX>

```

5.6 Delivery request (05)

This operation is used by the client (application) to retrieve a message sent to the client. The usage of this operation depends on the type of message centre and the type of application.

For the SMS Center, the usage depends on the application type.

The querying type of application must always poll for messages that use this delivery request operation.

For the receiving type of application, this operation is optional as normally messages are delivered immediately to the application that uses the deliver message operation (020). This operation can still be useful for querying the count of messages that are waiting for the application.

This operation cannot be used by send-only applications.

If the response to the delivery request with mode 1 or 2 is positive, one or more deliver message operations follow until all the messages are delivered.

Number	Delivery request parameter	Presence
068	Mode	O

Table 21 Delivery request parameter

Mode can have one of the following values (default value = 1):

- 0 Number of messages that are waiting. The response contains the number of messages that are waiting to be retrieved, no actual messages are transferred after this request.
- 1 Deliver one message. The oldest message is delivered.
- 2 Deliver all messages (receiving applications only). All messages are delivered.

Number	Positive response parameters
066	Number of messages waiting*

Table 22 Positive response parameters for delivery request

* Positive response contains the parameter only if the mode parameter in the request was used with value zero (0).

Number	Negative response parameters
900	Error code
901	Error text (optional)

Table 23 Negative response parameters for delivery request

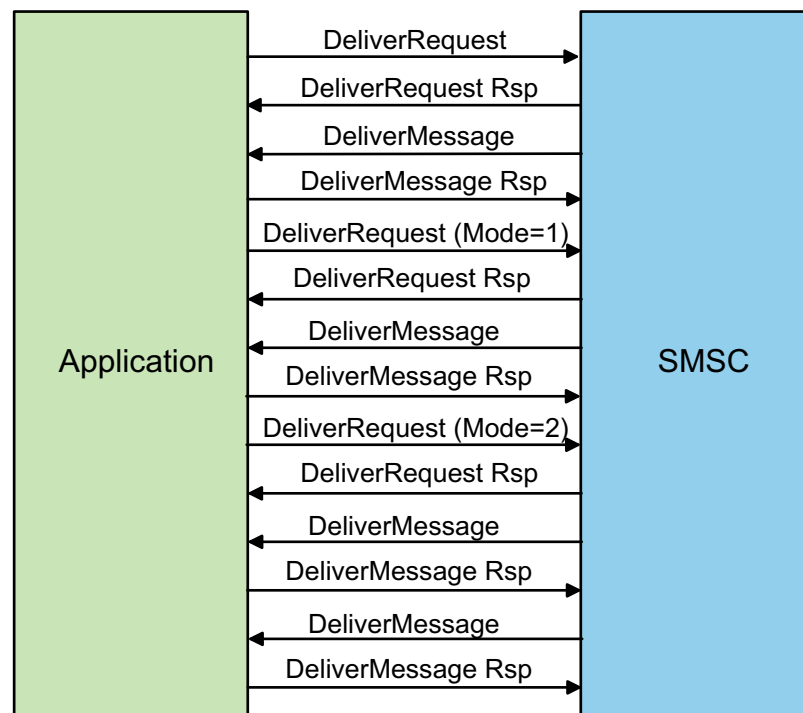


Figure 2 Flow of delivery request operations

Example: A querying application that uses the delivery request for retrieving messages

```

Delivery Request ----->
<STX>05:007<TAB><ETX>
<--- Positive response
<STX>55:007<TAB><chksum><ETX>

<----- Deliver message
<STX>20:002<TAB>021:DestAddr<TAB>060:ServiceCentreTimeStamp
<TAB>023:OrigAddr<TAB>033:Data<TAB><chksum><ETX>

Deliver message response ----->
    
```

```
<STX>70:002<TAB><ETX>
```

```
Delivery Request ----->
```

```
<STX>05:009<TAB><ETX>
```

```
<--- Positive response
```

```
<STX>55:009<TAB><checksum><ETX>
```

```
<----- Deliver message<STX>20:004<TAB>021:DestAddr<TAB>060:ServiceCentreTimeStamp
<TAB>023:OrigAddr<TAB>033:Data<TAB><checksum><ETX>
```

```
Deliver message response ----->
```

```
<STX>70:004<TAB>checksum<ETX>
```

Example: A querying or receiving application asks for the number of messages that are waiting

```
Delivery Request (mode=0) --->
```

```
<STX>05:011<TAB>068:0<TAB><ETX>
```

```
<--- Positive response
```

```
<STX>55:011<TAB>066:26<TAB><checksum><ETX>
```

5.7 Cancel message (06)

This operation is used to cancel a message sent earlier by the subscriber. It is possible to cancel more than one message with one operation. If the message has already been delivered to its destination, it cannot be cancelled.



When an application uses the `multi-instance mode = 2` setting, each instance of the application can cancel only messages that it has sent.

Cancelling can be disabled in the submit operation. Disabling is useful, for example, in cases where there are such messages to a certain destination that should not be cancelled, but the cancellation is made according to the destination address.

Number	Cancel message parameters	Presence
021	Destination address	O
059	Cancel mode	M
060	Service centre time stamp	O

Table 24 Cancel message parameters

The cancel mode parameter can have three different values:

- 0 Cancel all the messages with the same destination address. This is only valid for mobile-terminated (MT) messages.
- 1 Cancel all the sent messages. This is only valid for MT messages.
- 2 Cancel a message where the destination address and the time stamp match with a short message (SM) submitted earlier.

In mode 0, the destination address is mandatory. In mode 1, only the cancel mode parameter is needed, and in mode 2, all the three parameters are mandatory.

The positive response for cancel message does not contain any parameters.

Number	Negative response parameters
900	Error code
901	Error text (optional)

Table 25 Negative response parameters for cancel message

A cancel message response is positive even if the message cannot be cancelled. A positive response only means that the cancel request was received correctly by the SMS Center. SRs can be used for information on the actual success of the cancel message operation.

5.8 Deliver message (20)

This operation is used by the SMS Center to deliver a message to an application. Only receiving-type applications get this message.

Number	Deliver message parameters	Presence
021	Destination address	M
023	Originator address	M
026	Originated IMSI	O
028	Originated visited MSC address	O
030	Data coding scheme	O
032	User data header *	O
033	User data **	O
034	User data binary **	O
052	Protocol identifier	O
056	Status report request	O
060	Service centre time stamp	M
069	Service Center Address	O
070	Message reference	O

Table 26 Deliver message parameters

* The user data header (UDH) parameter is present only if the message contains a user data header.

** Either the user data (033) or user data binary (034) is used, depending on the value of the data coding scheme (030).

If the data coding scheme indicates that the data used the default GSM character set, the common user data parameter is used. For UCS2 or other binary data, the user data binary (034) parameter is used.

Other optional parameters are normally present, unless suppressed through the user interface profile.

The response for a deliver message does not need to contain any parameters, but parameter 900 (error code) is supported. See the following options:

- If the parameter is missing, the message is considered to be delivered successfully.
- If the value of the parameter is between range 700-799, the SMS Center handles the message as a failed delivery.

This functionality must be enabled in the user profile of the applications file with the `sm deliver nack enabled` parameter. For more information, see [s2aupsamplemx.cf](#) in *Configuration Files* (PDF).

The following values are defined:

- 700-709 delivery OK but pending
700 - delivery OK until the next network element
The SMS Center sends the originator a positive SR with log success indicator (LSI) 12, 'waiting for delivery'. This can be used, for example, with gateway type of applications.
- 710-729 delivery not OK, message level error
710 - generic failure
The SMS Center sends a negative SR.
711 - unsupported DCS
The SMS Center sends a negative SR.
712 - unsupported UDH
The SMS Center sends a negative SR.
- 730-750 delivery not OK, service level error
730 - unknown subscriber
The SMS Center sends a negative SR.

Error code values higher than 700 are visible in the CIMD2 operator session traces, but are not visible in event logs, and they are mapped as permanent errors.

- If the value of the parameter is not in the range 700-799, the SMS Center retries message delivery after a certain period of time.

Number	Negative response parameter
900	Error code

Table 27 Negative response parameter for delivery message

5.9 Deliver status report (23)

This operation is used by the SMS Center to send a status report (SR) that describes the delivery status of a previously submitted message. The generation of SRs for a particular message is requested in the submit operation.

The delivery of SRs to the application depends also on the type of application. Querying applications, connecting to an SMS Center, always have to request the delivery of an SR that uses the enquire message status operation (04). Receiving applications receive the SR whenever it is available.

After a successful delivery of the SR, the SR is removed from the SMS Center if it describes the final status of the message.

The information in the deliver SR message is the same as in the enquire message status response.

Number	Deliver status report parameters	Presence
021	Destination address	M
023	Originator address	O
060	Service centre time stamp	M
061	Status code	M
062	Status error code	O
063	Discharge time	M

Table 28 Deliver status report parameters

The response for deliver SR does not contain any parameters.

Example: Deliver status report

```

<--- Deliver Status Report
<STX>23:012<TAB>021:destAddr<TAB>060:scts
<TAB>061:statusCode<TAB>063:dis_time<TAB><chksum><ETX>
Response --->
<STX>73:012<TAB><chksum><ETX>
    
```

5.10 Set (08)

This operation is used by the application to change those parameters of the interface profile that it is allowed to change.

At the moment, only the password can be changed by applications.

Number	Set parameter
011	Password

Table 29 Set parameter

The positive response for set does not contain any parameters.

Number	Negative response parameters
900	Error code
901	Error text (optional)

Table 30 Negative response parameters for set

Example: Set

```

Set (password) --->
<STX>08:011<TAB>011:myNewpasWord<TAB><chksum><ETX>
<--- Response
<STX>58:011<TAB><chksum><ETX>
    
```

5.11 Get (09)

This operation is used by the application to retrieve specific parameters from the SMS Center.

In the get operation, parameter 500 is used in all get operations and the value of parameter 500 is the number of the parameter (or information) that the application requests.

Currently the only supported value for get parameters which returns the time of the SMS Center is 501.

Number	Positive response parameter
501	Time of the SMS Center

Table 31 Positive response parameter for get

The positive response contains the value parameter (or information) requested and the real value of that parameter.

The negative response follows if the parameter value is not available.

Number	Negative response parameters
900	Error code
901	Error text (optional)

Table 32 Negative response parameters for get

Example: Get

```
Get (system time)--->
<STX>09:009<TAB>500:501<TAB><checksum><ETX>
<----- response
<STX>59:009<TAB>501:990814154512<TAB><checksum><ETX>
```

5.12 Alive (40)

This operation can be used to check whether the link between the application and the SMS Center is still alive. The receiving entity sends an acknowledgement back to the originator if the alive operation is received correctly.

The SMS Center does not initiate the alive operation, so it is the responsibility of the application to keep the link alive by invoking the alive operation periodically. The SMS Center responds with an acknowledgement if it receives the alive operation correctly from an application.

The alive operation does not need any parameters. The positive response for alive does not contain any parameters.

Example: Application-originated alive request

```
Alive request ----->
<STX>40:009<TAB><Chksum><ETX>
<----- response
```


<STX>90:009<TAB><Checksum><ETX>

5.13 Additional operations

This section lists the additional response packages used in certain error situations.

5.13.1 General error response (98)

This operation is used by the SMS Center to respond to illegal, unexpected, or unsupported operations sent by the application.

Number	Operation parameters
900	Error code
901	Error text (optional)

Table 33 Operation parameters

Example: General error response

```
Request ---->
<STX>07:007<TAB><checksum><ETX>
<---- response
<STX>98:007<TAB>900:ErrorCode<TAB><checksum><ETX>
```

5.13.2 NACK (99)

The NACK (negative acknowledgement) operation is used to reject an operation due to an incorrect checksum or an incorrect sequence number. The NACK operation causes retransmission of the message. The packet sequence number in a NACK message is always the expected sequence number.

There are no operation parameters.

Example: Corrupted message retransmission

```
SME sends ----->
<STX>03:007<TAB>021:daddr<TAB>033:data<TAB>3A<ETX>
SMS Center Receives
<STX>03:007<TAB>021:daddr<TAB>033:XXXX<TAB>3A<ETX>

<----- response with the nack
<STX>99:007<TAB><checksum><ETX>

SME retransmits ----->
<STX>03:007<TAB>021:daddr<TAB>033:data<TAB>3A<ETX>

<----- SMS Center response
<STX>53:007<TAB>021:daddr<TAB>060:scts<TAB><CS><ETX>
```

Example: Wrong packet number

```
SME sends unexpected packet number ----->
<STX>03:001<TAB>021:daddr<TAB>033:Data<TAB><CS><ETX>
<----- SMS Center responds with nack
<STX>99:009<TAB><CS><ETX>
```

```
SME recovers ---->
<STX>03:009<TAB>021:daddr<TAB>033:Data<TAB><CS><ETX>
```

```
<----- SMS Center responds
<STX>53:009<TAB>021:daddr<TAB>060:60:scts<TAB><CS><ETX>
```

The SMS Center expects a packet number 007, but the client sends something else. After the NACK, the client recovers and normal operation continues.

6 Windowing for applications

The windowing for applications feature enables the CIMD2 applications to initiate more than one operation before receiving responses. The feature also activates multiple short message (SM) and status report (SR) deliveries from the application server (ASE) to the application, which application can acknowledge the deliveries preferably in the received order.

This feature is useful, for example, when the application connects to the ASE through a slow link. In this case the application can speed up the transactions by submitting more than one SM before waiting for submit responses.

6.1 Windowing scenarios

There are several scenarios between the application and the ASE.

6.1.1 Submitting more than one short message from application

Suppose X is the window size allowed for the considered application. The application is allowed to submit up to X SMs to the ASE before it waits for the responses. The ASE ignores any SMs submitted with a sequence number outside the window range and a NACK (negative acknowledgement) is returned. So there is no retransmission of responses when windowing is enabled. This behaviour is enabled only when the application logs in with parameter 019. The application is forced to use the sequence numbers in strict increasing order.

If an application submits with a window size 3 and if 007 is the current sequence number for submit, the following scenario is possible:

ASE	Direction	Application
	< — — —	007:<submit>
007:<submit response>	— — —>	
	< — — —	005:<submit>
— — — Ignore (NACK) — — —		
	< — — —	011:<submit>
— — — Ignore (NACK) — — —		
	< — — —	009:<submit>
	< — — —	011:<submit>
	< — — —	013:<submit>
009:<submit response>	— — —>	
011:<submit response>	— — —>	
013:<submit response>	— — —>	

Table 34 Windowing scenario 1

ASE delivers more than one SM/SR to application

If an application has logged in with a window size X and the ASE has accepted it, the ASE is entitled to deliver up to X SMs or SRs to the application before waiting for the

response from the application. This is valid only if the application is configured as auto deliver. If the delivery is not acknowledged during the delivery timeout period, the ASE retransmits the packet. The application can acknowledge the delivery packets that it has received in any order, but the preferred way is to acknowledge them in the order they are received.

If the ASE retrieves messages from the message poller, the ASE uses only a window size of 1. The scenario can be explained with an application that has a window size of 3, see the following table.

ASE	Direction	Application
000:<SM delivery>	— — —>	
002:<SM delivery>	— — —>	
004:<SM delivery>	— — —>	
	< — — —	000:<SM ack>
006:<SM delivery>	— — —>	
	< — — —	002:<SM ack>
	< — — —	006:<SM ack>
...		
...		
Time out for 004 — retransmit	— — —>	
004:<SM delivery>	< — — —	004:<SM ack>

Table 35 Windowing scenario 2

6.2 Windowing restrictions

The CIMD2 windowing does not distinguish between different operations initiated by the application. This means that an application can have X outstanding operations (submit, message count request, cancel, and so on) if it has been granted a window size of X.

All the windowing features are enabled if the application logs with parameter 019. If not, submit and delivery behaviour is the same as in the previous version of CIMD. But an application can use windowing only after it has received a login response from the ASE.

For polling applications, it is recommended not to use a window size higher than 1. This is because more than one simultaneous poll-request for an SM can cause duplicate deliveries.

7 CIMD parameters

7.1 Parameter types

The parameter type selects the allowed values of characters in CIMD2 messages.

Integer (int)

Integer is the most common type of parameter. The allowed values are the ASCII representation of the digits 0 to 9. This means that only decimal values from 48 to 57 are allowed.

Example: Integer

```
064:3<tab>050:167<tab>030:0<tab>055:1<tab>056:63
```

Address (addr)

The Address parameter type is used for passing GSM addresses to and from the SMS Center. This field can contain digits like the integer field, but also some other characters. The accepted characters depend on the configuration of the address conversion done in the SMS Center. The ASCII representation of '+' is currently allowed. This means that, besides the values 48 to 57 (digits 0 to 9), value 43 ('+') is also allowed.

Example: Address

```
021:+35812345678<tab>023:13131
```

Hexadecimal (hex)

The Hexadecimal parameter type is used for passing binary data to and from the SMS Center. This field can contain the ASCII representation of the digits 0 to 9 and the ASCII representation of the characters 'A' to 'F' and 'a' to 'f'. This means that the following decimal values are allowed: 48 to 57 (0 to 9); 65 to 70 ('A' to 'F') and 97 to 102 ('a' to 'f') in the CIMD2 parameters of this type.

Example: Sending six bytes of binary data

```
034:0500032a0301<tab>
```

User data (ud)

The User data parameter type is used to exchange user data between the SMS Center and an application. The application can use most of the printable ASCII character set and it can combine them to select the characters of the default GSM character set. The exact configuration of this conversion is selected by the user profile. For more information, see the default character conversion table in [Default character conversion](#) in *CIMD Interface Specification* (PDF).

The following example shows the first few characters of the default GSM character set that can be entered using the default character conversion over an 8-bit wide link using a system with the ISO-Latin character set.

Example: Default character conversion over 8-bit wide link

```
033:@£$¥èéùìòÇ
```

which is from the iso latin1 character set.

```
48,51,51,58,64,163,36,165,232,233,249,236,242,199
```

For a 7-bit link, only values under 128 can be used. The following example shows how to enter the same data as before (see the previous example) from a system with a different character set.

Example: Default character conversion over 7-bit link

```
033:_a0_L-$_Y-_e`_e"_u`_i`_o`_C,
```

String (str)

The String parameter type is used to exchange general printable characters from the ASCII character table. Parameters of this type can have the decimal values from 32 (space) to 126 ('~').

Example: String

```
010:MyUserId<tab>011:my()<pass;+\wo%=rd<tab>
```

7.2 Parameters

The table below specifies the CIMD parameters and their values.


Name	ID	Maximum length	Type	Values	Description
User identity	010	32	String	-	Identity used for login. Maximum 32 characters. Spaces are not allowed.
Password	011	32	String	-	Password used for login. Maximum 32 characters. Spaces are not allowed.
Sub-address	012	3	Integer	0-9	Defines a unique index for application instance. This is useful for correct delivery of SRs when multiple instances of the same application are connected.
Window size	019	3	Integer	1-128	Defines the window size used by the application for submitting messages.
Destination address	021	20	Address	-	Destination address in the GSM network. The prefix '+' indicates address type 145, which means international address. This parameter is valid only for application-terminated (AT) type of messages.  In the SubmitSR operation, multiple destination addresses are not allowed.
Originating address	023	20	Address	-	Originating address. This value is appended to the prefix in the ASE subscriber database. The total length can be maximum 20 characters. The '+' character can be used in front of the originating address in application-originated (AO) and AT messages to set the corresponding type of number to International (145). Note that the '+' character is only allowed in combination with the 'free submit originator address' feature.

Table 36 Parameters


Name	ID	Maximum length	Type	Values	Description
Originating IMSI	026	20	Address	-	Originating International Mobile Subscriber Identity number.
Alphanumeric originating address	027	11	String	-	Originating address in alphanumeric format. This parameter is a string of maximum 11 characters, where each character byte can be ASCII alphabet 32-126, and the characters \$ @] ` } are not allowed.  The characters [\ ^ { ~ are acceptable as input and displayed as Ä Ö Ü ä ö ü in phone.
Originated visited MSC address	028	20	Address	-	The ISDN number of the serving MSC network element of the originating subscriber.
Data coding scheme	030	3	Integer	0-255	As defined in GSM 03.38.
User data header	032	280-320	Hexadecimal	-	This is an optional part of the user data of the transferred message. The maximum length of the header is 140 bytes. The maximum length of the user data header is reduced by the length of the user data or the user data binary. 140 bytes result in a coded length of 280 characters.
User data	033	480-546	User data	-	This parameter can be used to transfer the user data of the short message (SM) by using the default character set. Special 3-character sequences can be used to select, for example, some Greek characters. The maximum length of the user data in the message can be maximum 160 septets. This corresponds to 140 octets or bytes.
User data binary	034	280-320	Hexadecimal	-	This field can be used to transfer the user data of the SM. As it uses the simple hexadecimal coding, any bit pattern can be sent to another application. By selecting an appropriate value for the data coding scheme (030) binary data or UCS2 user data can be transferred. The maximum length of the user data binary is 140 bytes. This maximum length is reduced by the length of the user data header (032).

Table 36 Parameters (Cont.)

Name	ID	Maximum length	Type	Values	Description
More messages to send	044	1	Integer	0 or 1	This parameter indicates if there are more messages to the same destination immediately after this message. This allows faster delivery of many messages to the same destination. Values: 0 – No subsequent messages 1 – More messages will follow NOTE! The value of this parameter turns to 0 if one of the <code>first delivery time relative (053)</code> or <code>first delivery time absolute (054)</code> parameters appears in the submit.
Validity period relative	050	3	Integer	-1-255	The length of the validity period of the message, counted from the time the SMS Center receives the message. The value is converted to a time period according to the <i>GSM 03.40 specification</i> . This parameter cannot be combined with the <code>validity period absolute (051)</code> parameter.
Validity period absolute	051	12	Integer	-	The absolute termination time of the validity period of the message; the value consists of year, month, day, hour, minute, second in the 'yymmddhmmSS' format. This parameter cannot be combined with the <code>validity period relative (050)</code> parameter.
Protocol identifier	052	3	Integer	0-255	The protocol identifier (PID) can be used to control GSM phase 2 features, such as 'replace message'. Setting the PID to 65 (decimal) sets the message type to 'Replace Message Type 1', adding 1...6 gives the replace message types 2 to 7, as defined in <i>GSM 03.40</i> . This parameter can also select the routing to applications instead of GSM mobile stations, as defined in the SMS Center routing table.
First delivery time relative	053	3	Integer	-1-255	This parameter selects the time period waited before the first delivery attempt of the message is made. The value is converted like the <code>validity period relative</code> .
First delivery time absolute	054	12	Integer	-	Time for the first delivery attempt of the message. Representation in the 'yymmddhmmSS' format.
Reply path	055	1	Integer	0 or 1	This parameter selects whether the reply path feature is enabled (1) or not (0).

Table 36 Parameters (Cont.)

Name	ID	Maximum length	Type	Values	Description
Status report request	056	3	Integer	0-127	<p>Defines in what cases the SR is returned.</p> <p>Value of the parameter is the sum of the cases where automatic SR is produced. For example, value 62 means that SR is created for all events except for 'first temporary result'.</p> <p>Possible flags for SMS Center connections:</p> <ul style="list-style-type: none"> 1 – temporary error 2 – validity period expired 4 – delivery failed 8 – delivery successful 16 – message cancelled 32 – message deleted by the operator 64 – first temporary result 128 – reserved for SMS Center internal use <p>Resulting in SR request values from 0-127, and no other values should be used.</p>
Cancel enabled	058	1	Integer	0 or 1	<p>This parameter selects whether the submitted mobile-terminated (MT) message can later be cancelled (1) or not (0).</p> <p>AT messages can be cancelled regardless of this setting.</p>
Cancel mode	059	1	Integer	0, 1 or 2	<p>Cancel mode:</p> <ul style="list-style-type: none"> 0 – Cancel all messages with the same destination address. 1 – Cancel all sent messages. 2 – Cancel a message where the destination address and the time stamp match with a message submitted earlier.
Service centre time stamp	060	12	Integer	-	<p>Time of message arrival at the SMS Center.</p> <p>In the SMS Center connection the format is 'yymmddhhmmss'.</p>

Table 36 Parameters (Cont.)

Name	ID	Maximum length	Type	Values	Description
Status code	061	2	Integer	See Description.	Status of the message delivery. Passed to the application in the enquire message status response and deliver SR messages. The following values apply to SMS Center connections: 0 – no status available (cannot be used for Submit SR) 1 – in process 2 – validity period expired 3 – delivery failed 4 – delivery successful 5 – no response 6 – last no response 7 – message cancelled 8 – message deleted 9 – message deleted by cancel i status code represents the latest status of the whole message handling flow for one short message, include short message itself and status reports it triggered. In order to get the status of the original short message, reference to “status error code” is recommended.
Status error code	062	3	Integer	0-999	Detailed error code of the message delivery status. The values are listed in Status Error Code values in <i>CIMD Interface Specification</i> (PDF).
Discharge time	063	12	Integer	-	Time of the last delivery attempt in the 'yymmddhhmMSS' format.
Tariff class	064	2	Integer	0-99	Defines the tariff class of the message.
Service description	065	2	Integer	0-99	Defines the service description of the message that can be used for billing.
Message count	066	3	Integer	0-999	Number of messages waiting to be retrieved.
Priority	067	1	Integer	1-9	Priority of the message. Lower value means higher priority.
Delivery request mode	068	1	Integer	0, 1 or 2	Selects the mode in the delivery request operation.
Service center address	069	20	Address	-	The ISDN number of an SMS Center element.
Message reference	070	3	Integer	0-255	Message reference ID, which identifies a message on the MS. It is used to connect SRs with the corresponding SMs.

Table 36 Parameters (Cont.)

Name	ID	Maximum length	Type	Values	Description
IP address	071	15 bytes	String (in IP address format)	-	This parameter can be specified in the login request to provide application's IP address other than its socket connection address.
Get parameter	500	3	Integer	501-999	The value is the parameter number or information item that the application wants to retrieve. Currently only value 501 is supported.
SMS Center time	501	12	Integer	-	The local time of the SMS Center in the 'ymmddhhmMSS' format.
Error code	900	3	Integer	0-999	The identifier of the error. The error codes are listed in Error codes in <i>CIMD Interface Specification</i> (PDF).
Error text	901	64	String	-	Text describing the error. These merely help so that the client application developer does not have to look up the coded error.

Table 36 Parameters (Cont.)

8 Error codes

This section describes general error codes and Status Error Code.

8.1 General error codes

The following table lists the values of the parameter 900 'error code' together with the error texts.

Error code	Error text
0	No error
1	Unexpected operation
2	Syntax error
3	Unsupported parameter error
4	Connection to SMS Center lost
5	No response from SMS Center
6	General system error
7	Cannot find information
8	Parameter formatting error
9	Requested operation failed
10	Temporary congestion error
LOGIN error codes:	
100	Invalid login
101	Incorrect access type
102	Too many users with this login ID
103	Login refused by SMS Center
104	Invalid window size
105	Windowing disabled
106	Virtual SMS Center-based barring
107	Invalid subaddr
108	Alias account, login refused
SUBMIT MESSAGE and SUBMIT SR error codes:	
300	Incorrect destination address
301	Incorrect number of destination addresses
302	Syntax error in user data parameter
303	Incorrect bin/head/normal user data parameter combination
304	Incorrect dcs parameter usage
305	Incorrect validity period parameter usage
306	Incorrect originator address usage

Table 37 Parameter error codes and error texts

Error code	Error text
307	Incorrect PID parameter usage
308	Incorrect first delivery parameter usage
309	Incorrect reply path usage
310	Incorrect status report request parameter usage
311	Incorrect cancel enabled parameter usage
312	Incorrect priority parameter usage
313	Incorrect tariff class parameter usage
314	Incorrect service description parameter usage
315	Incorrect transport type parameter usage
316	Incorrect message type parameter usage
318	Incorrect MMS parameter usage
319	Incorrect operation timer parameter usage
320	Incorrect dialogue ID parameter usage
321	Incorrect alpha originator address usage
322	Invalid data for alphanumeric originator
323	Online closed user group rejection
324	Licence exceeded
326	Submit SR is disabled in the user profile thus it cannot be used
327	Incorrect or missing message reference
328	Incorrect SC address was specified
329	SCTS is invalid
330	Incorrect status code
332	Incorrect discharge time specified
ENQUIRE MESSAGE STATUS error codes:	
400	Incorrect address parameter usage
401	Incorrect scts parameter usage
DELIVERY REQUEST error codes:	
500	Incorrect scts parameter usage
501	Incorrect mode parameter usage
502	Incorrect parameter combination
CANCEL MESSAGE error codes:	
600	Incorrect scts parameter usage
601	Incorrect address parameter usage
602	Incorrect mode parameter usage
603	Incorrect parameter combination
DELIVER MESSAGE error codes:	

Table 37 Parameter error codes and error texts (Cont.)

Error code	Error text
700	Delivery OK/ waiting for delivery
710	Generic failure
711	Unsupported DCS
712	Unsupported UDH
730	Unknown subscriber
SET error codes:	
800	Changing password failed
801	Changing password not allowed
GET error codes:	
900	Unsupported item requested

Table 37 Parameter error codes and error texts (Cont.)

8.2 Status Error Code values

The following table lists the values and meanings of the Status Error Code (062) parameter for SMS Center connections.

Value	Meaning
0	No error
1	Unknown subscriber
9	Illegal subscriber
11	Teleservice not provisioned
13	Call barred
15	OCUG reject
19	No SMS support in mobile station (MS)
20	Error in MS
21	Facility not supported
22	Memory capacity exceeded
29	Absent subscriber
30	MS busy for mobile-terminated (MT) SMS
36	Network/Protocol failure
44	Illegal equipment
60	No paging response
61	GMSC congestion
63	HLR timeout
64	MSC/SGSN timeout
70	SMRSE/TCP error

Table 38 Status Error Code parameter for SMS Center connections

Value	Meaning
72	MT congestion
75	GPRS suspended
80	No paging response through MSC
81	IMSI detached
82	Roaming restriction
83	Deregistered in HLR for GSM
84	Purged for GSM
85	No paging response through SGSN
86	GPRS detached
87	Deregistered in HLR for GPRS
88	The MS purged for GPRS
89	Unidentified subscriber through MSC
90	Unidentified subscriber through SGSN
112	Originator missing credit on prepaid account
113	Destination missing credit on prepaid account
114	Error in prepaid system

Table 38 Status Error Code parameter for SMS Center connections (Cont.)

9 Character conversion for user data

The user data can use the default character set and the characters from the extension table defined in the *3GPP TS 23.038 Specification*. The character conversion is made based on these character tables.

9.1 Default character conversion

The following table shows how the characters of the default character set table are converted in short message (SM) user data.

The three columns on the left show the value, symbol, and name of the character of the 7-bit default GSM character set.

The fourth column shows the possible character of the ISO-Latin character set and the decimal value in brackets. Computers using another character table can also send the value from this column, even though it looks like a different character on their system. The fifth column shows the combination of ISO-Latin characters which results in the intended GSM character. The characters used in this column also match the corresponding characters of the common ASCII table.

GSM character			Iso-Latin	Special combination
0	@	COMMERCIAL AT	@ (64)	_Oa
1	£	POUNDS STERLING		_L-
2	\$	DOLLAR SIGN	\$ (36)	
3	¥	YEN		_Y-
4	è	e WITH GRAVE ACCENT		_e`
5	é	e WITH ACUTE ACCENT		_e'
6	ù	u WITH GRAVE ACCENT		_u`
7	ì	i WITH GRAVE ACCENT		_i`
8	ò	o WITH GRAVE ACCENT		_o`
9	ç	C WITH CEDILLA		_C,
10		LINE FEED	(10)	
11	ø	O WITH SLASH		_O/
12	ø	o WITH SLASH		_o/
13		CARRIAGE RETURN	(13)	
14	Å	A WITH RING] (93)	_A*
15	å	a WITH RING	} (125)	_a*
16	Δ	GREEK ALPHABET DELTA		_gd
17	_	UNDERSCORE		_--
18	Φ	GREEK ALPHABET PHI		_gf
19	Γ	GREEK ALPHABET GAMMA		_gg
20	Λ	GREEK ALPHABET LAMBDA		_gl

Table 39 Default character conversion for user data

GSM character			Iso-Latin	Special combination
21	Ω	GREEK ALPHABET OMEGA		_go
22	Π	GREEK ALPHABET PI		_gp
23	Ψ	GREEK ALPHABET PSI		_gi
24	Σ	GREEK ALPHABET SIGMA		_gs
25	Θ	GREEK ALPHABET THETA		_gt
26	Ξ	GREEK ALPHABET XI		_gx
27		RESERVED		_XX
28	Æ	AE DIPHTHONG		_AE
29	æ	ae DIPHTHONG		_ae
30	ß	GERMAN DOUBLE-S		_ss
31	É	E WITH ACUTE ACCENT		_E'
32		SPACE	(32)	
33	!	EXCLAMATION MARK	! (33)	
34	"	QUOTATION MARK	" (34)	_qq
35	#	NUMBER SIGN	# (35)	
36	¤	CURRENCY SYMBOL		_ox
37	%	PERCENT SIGN	% (37)	
38	&	AMPERSAND	& (38)	
39	'	APOSTROPHE	' (39)	
40	(LEFT PARENTHESIS	((40)	
41)	RIGHT PARENTHESIS) (41)	
42	*	ASTERISK	* (42)	
43	+	PLUS SIGN	+ (43)	
44	,	COMMA	, (44)	
45	-	HYPHEN	- (45)	
46	.	FULL STOP (PERIOD)	. (46)	
47	/	SOLIDUS (SLASH)	/ (47)	
48	0	DIGIT ZERO	0 (48)	
49	1	DIGIT ONE	1 (49)	
50	2	DIGIT TWO	2 (50)	
51	3	DIGIT THREE	3 (51)	
52	4	DIGIT FOUR	4 (52)	
53	5	DIGIT FIVE	5 (53)	
54	6	DIGIT SIX	6 (54)	
55	7	DIGIT SEVEN	7 (55)	
56	8	DIGIT EIGHT	8 (56)	

Table 39 Default character conversion for user data (Cont.)

GSM character			Iso-Latin	Special combination
57	9	DIGIT NINE	9 (57)	
58	:	COLON	: (58)	
59	;	SEMICOLON	; (59)	
60	<	LESS-THAN SIGN	< (60)	
61	=	EQUALS SIGN	= (61)	
62	>	GREATER-THAN SIGN	> (62)	
63	?	QUESTION MARK	? (63)	
64	¡	INVERTED !		_!!
65	A	CAPITAL LETTER A	A (65)	
66	B	CAPITAL LETTER B	B (66)	
67	C	CAPITAL LETTER C	C (67)	
68	D	CAPITAL LETTER D	D (68)	
69	E	CAPITAL LETTER E	E (69)	
70	F	CAPITAL LETTER F	F (70)	
71	G	CAPITAL LETTER G	G (71)	
72	H	CAPITAL LETTER H	H (72)	
73	I	CAPITAL LETTER I	I (73)	
74	J	CAPITAL LETTER J	J (74)	
75	K	CAPITAL LETTER K	K (75)	
76	L	CAPITAL LETTER L	L (76)	
77	M	CAPITAL LETTER M	M (77)	
78	N	CAPITAL LETTER N	N (78)	
79	O	CAPITAL LETTER O	O (79)	
80	P	CAPITAL LETTER P	P (80)	
81	Q	CAPITAL LETTER Q	Q (81)	
82	R	CAPITAL LETTER R	R (82)	
83	S	CAPITAL LETTER S	S (83)	
84	T	CAPITAL LETTER T	T (84)	
85	U	CAPITAL LETTER U	U (85)	
86	V	CAPITAL LETTER V	V (86)	
87	W	CAPITAL LETTER W	W (87)	
88	X	CAPITAL LETTER X	X (88)	
89	Y	CAPITAL LETTER Y	Y (89)	
90	Z	CAPITAL LETTER Z	Z (90)	
91	Ä	A WITH DIERESIS	[(91)	_A"
92	Ö	O WITH DIERESIS	\ (92)	_O"

Table 39 Default character conversion for user data (Cont.)

GSM character			Iso-Latin	Special combination
93	Ñ	N WITH TILDE		_N~
94	Ü	U WITH DIERESIS	^ (94)	_U"
95	§	SECTION MARK		_so
96	¿	INVERTED ?		_??
97	a	SMALL LETTER a	a (97)	
98	b	SMALL LETTER b	b (98)	
99	c	SMALL LETTER c	c (99)	
100	d	SMALL LETTER d	d (100)	
101	e	SMALL LETTER e	e (101)	
102	f	SMALL LETTER f	f (102)	
103	g	SMALL LETTER g	g (103)	
104	h	SMALL LETTER h	h (104)	
105	i	SMALL LETTER i	i (105)	
106	j	SMALL LETTER j	j (106)	
107	k	SMALL LETTER k	k (107)	
108	l	SMALL LETTER l	l (108)	
109	m	SMALL LETTER m	m (109)	
110	n	SMALL LETTER n	n (110)	
111	o	SMALL LETTER o	o (111)	
112	p	SMALL LETTER p	p (112)	
113	q	SMALL LETTER q	q (113)	
114	r	SMALL LETTER r	r (114)	
115	s	SMALL LETTER s	s (115)	
116	t	SMALL LETTER t	t (116)	
117	u	SMALL LETTER u	u (117)	
118	v	SMALL LETTER v	v (118)	
119	w	SMALL LETTER w	w (119)	
120	x	SMALL LETTER x	x (120)	
121	y	SMALL LETTER y	y (121)	
122	z	SMALL LETTER z	z (122)	
123	ä	a WITH DIERESIS	{ (123)	_a"
124	ö	o WITH DIERESIS	(124)	_o"
125	ñ	n WITH TILDE		_n~
126	ü	u WITH DIERESIS	~ (126)	_u"
127	à	a WITH GRAVE ACCENT		_a`

Table 39 Default character conversion for user data (Cont.)

9.2 Extension character conversion

The following table shows how the characters of the extension table defined in *3GPP TS 23.038 Specification* are converted in short message (SM) user data.

In the CIMD user data parameter, the special combination character `_XX` (named Reserved) is reserved as an escape to an extension of the GSM 7-bit default alphabet table. If a receiving mobile station (MS) does not understand the meaning of this escape mechanism, it must display it as a space character.

When defining the CIMD Reserved escape character (i.e. `_XX`) and the second CIMD character, type them immediately after each other (no space between them) in the CIMD user data parameter.

Symbol	Character name	CIMD special combination	Name of combination
	Vertical bar	<code>_XX_!</code>	Reserved, Inverted !
^	Caret	<code>_XX_λ</code>	Reserved, Greek Alphabet Lambda
<euro symbol>	Euro symbol *	<code>_XXe</code>	Reserved, Small Letter e
{	Opening brace	<code>_XX(</code>	Reserved, Left Parenthesis
}	Closing brace	<code>_XX)</code>	Reserved, Right Parenthesis
	Page break	<code>_XXascii(10)</code>	Reserved, Line Feed ascii(10)
[Opening bracket	<code>_XX<</code>	Reserved, Less Than Sign
]	Closing bracket	<code>_XX></code>	Reserved, Greater Than Sign
~	Tilde	<code>_XX=</code>	Reserved, Equals Sign
\	Backslash	<code>_XX/</code>	Reserved, Slash

Table 40 Extension character conversion for user data

* The euro symbol is defined in the 7-bit default alphabet extension table in the *3GPP TS 23.038 Specification*. The code value is the one used for character 'e' in the 7-bit default GSM character set. Therefore, a receiving entity that is incapable of displaying the euro currency symbol displays the character 'e' instead.

The euro symbol is displayed correctly only on mobile terminals that support it.

9.3 Traffic direction specific character conversion

It is possible that some language-specific special characters are not converted correctly in application-terminated (AT) messages when the default conversion files are used. This happens if the same character conversion table is used for both application-originated (AO) and application-terminated (AT) direction. The ASE consults first the default character conversion table and then the special character conversion table in the AT direction. If the ASE finds a value in the default character conversion table, it sends the value to the application. However, if the expected character should have been taken from the special character conversion table, the conversion produces an unexpected result.

To avoid the misconversions, it is possible to define different character conversion tables to be used in AO and in AT directions instead of the default conversion files. This is done by enabling the direction-specific character conversion files in the user profile file in the following way:

```
#Character conversion tables
#character table file=/etc/opt/nokia/smsc/conf/xtm_isolat_mx.cf
#special character table file=/etc/opt/nokia/smsc/conf/xtm_specia_mx.cf
AO character table file=/etc/opt/nokia/smsc/conf/xtm_isolatmx.cf
AO special character table file=/etc/opt/nokia/smsc/conf/xtm_speciamx.cf
AT character table file=/etc/opt/nokia/smsc/conf/xtm_ATisolatmx.cf
AT special character table file=/etc/opt/nokia/smsc/conf/xtm_ATspeciamx.cf
```

In case some characters still cause problems in translation, it is possible to edit the AT file contents and to comment out those characters that are not converted correctly as well.



You must define all the four settings for AO and AT character conversions. Otherwise, the ASE uses the default conversion files.

The AO direction can still use the default files, there is no need to customise them.

Remember to comment out the default files.

For more information, see the configuration instructions in [Adding traffic direction-specific character conversion tables](#) in *External Applications Configuration Guide* (PDF).

Sending a UCS2 encoded message (binary data)

A short text 'test' that has been encoded to Unicode UCS2 is sent to an MS. The main components are the UDB and the DCS.

The DCS is coded as defined in *ETSI GSM 03.38 Specification*. For UCS2, it is set to 8.

The total message in terms of CIMD2 protocol:

```
<stx>03:003<tab>
021:1234567890<tab>
030:8<tab>
034:0074006500730074<tab><etx>
```

Sending a concatenated message containing text data with MMS hinting

A message containing 180 characters of text is sent from an application to an MS. The application is responsible for performing the concatenation before sending the data to the SMS Center. The MS then reassembles the original message based on the concatenation information delivered in the UDH.

The main components are the UDH, user data (UD), and more messages to send (MMS).

The maximum length of one message is 160 characters. The UDH that is required to indicate concatenation is also considered as part of the UD when calculating the message length. This means that the first part can contain: maximum length - (udh length * 8bits / 7bits) characters. In this example the first part has $160 - (6 * 8 / 7) = 153$ characters.

The more messages to send parameter indicates that there are more messages to the same destination immediately after this message. This allows faster delivery of many messages to the same destination. The first CIMD2 message has the parameter set to 1, and the second one to 0.

The UDH (in this example: 050003000201 and 050003000202) indicates that this is a concatenated message.

Number	UDH information
05	Length of the UDH
00	IEI (concatenated short messages, 8 bit reference number)
03	IEDL
00	Modulo 256 reference number
02	Total amount of message parts
01/02	Sequence number of concatenated message

The messages in terms of CIMD2 protocol:

```
<stx>03:003<tab>
021:1234567890<tab>
032:050003000201<tab>
033:<153 characters of user data><tab>
044:1<tab><etx>

<stx>03:005<tab>
021:1234567890<tab>
```



```
032:050003000202<tab>  
033:<27 characters of user data><tab>  
044:0<tab><etx>
```

10.2 Example sessions

10.2.1 Example session for a send-only application

The example session given here contains the login, submit message, and logout operations.



In the submit operation example, only parameters destination address, validity period, and user data are given by the application. Packets are split at special characters, for example <TAB> for printing purposes only.

```
SME <--> SMSC
Connect <-->
Greeting <--
Login -->
<STX>01:001<TAB>010:MyUserId<TAB>011:MySecretPassWOrD<TAB><CS><ETX>
Login response <--
<STX>51:001<TAB><CS><ETX>

Submit -->
<STX>03:003<TAB>021:1234567890<TAB>050:167<TAB>033:Hi
there !<TAB> <CS><ETX>

Submit response (positive) <--
<STX>53:003<TAB>021:1234567890<TAB>060:971111131245<TAB><CS><ETX>

Submit -->
<STX>03:005<TAB>050:167<TAB>033:Hi
there ! How are you doing ?<TAB><CS><ETX>

Submit response (negative) //destination was missing!! <--
<STX>53:005<TAB>900:301<TAB><CS><ETX>

Enquiry message status-->
<STX>04:007<TAB> 060:971111131245<TAB>021:1234567890<TAB><CS><ETX>

Enquiry message status response <--
<STX>54:007<TAB>021:1234567890<TAB>060:971111131245<TAB>061:0<TAB->063:971111131245<TAB><CS><ETX>

Disconnect <-->
```

10.2.2 Example session for querying application

```
SME <--> SMSC
Connect <-->
Greeting <--
Login -->
<STX>01:001<TAB>010:MyUserId<TAB>011:MySecretPassWOrD<TAB><CS><ETX>
Login response <--
<STX>51:001<TAB><CS><ETX>

Submit -->
<STX>03:003<TAB>021:1234567890<TAB>050:167<TAB>033:Hi
there !<TAB><CS><ETX>

Submit response (positive) <--
<STX>53:003<TAB>021:1234567890<TAB>060:971111131245<TAB><CS><ETX>
```

Delivery Request Message -->
<STX>05:005<TAB>068:1<TAB><CS><ETX>

Delivery Request Response (positive) <-->
<STX>55:005<TAB><CS><ETX>

Deliver Message <-->
<STX>20:000<TAB>021:1234567890<TAB>023:987654321<TAB>060:971111121245 033:Call
me.<TAB><CS><ETX>

Deliver Message Response -->
<STX>70:000<TAB><CS><ETX>

Disconnect <-->

10.2.3 Example session for receiving application

SME <--> MC
Connect <-->
Greeting <-->
Login -->
<STX>01:001<TAB>010:MyUserId<TAB>011:MySecretPassWorD<TAB><CS><ETX>
Login response <-->
<STX>51:001<TAB><CS><ETX>

Submit -->
<STX>03:003<TAB>021:1234567890<TAB>050:167<TAB>033:Hi
there!<TAB><CS><ETX>

Deliver Message //delivery before submit response
received <-->
<STX>20:000<TAB>021:9876<TAB>023:1234567890<TAB>060:971111080045<TAB->033:Call
me!<TAB>052:0<TAB>030:0<TAB><CS><ETX>

Deliver Message Response -->
<STX>70:000<TAB><CS><ETX>

Submit response (positive) <-->
<STX>53:003<TAB>021:1234567890<TAB>060:971111131245<TAB><CS><ETX>

Deliver Message <-->
<STX>20:002<TAB>021:98765<TAB>023:1234567890<TAB>060:971111121245<TAB>033:Why
dont you call me<TAB>052:0 030:0<TAB><CS><ETX>

Deliver Message Response -->
<STX>70:002<TAB><CS><ETX>

Deliver StatusReport <-->
<STX>23:005<TAB>021:1234567890<TAB>060:971111131245<TAB>061:3<TAB>063:971111131354<TAB><CS><ETX>

Deliver StatusReport Response -->
<STX>73:005<TAB><CS><ETX>

Disconnect <-->