SMS 365 HTTP Technical Specification

Version 4.2.2 – April 2021

TABLE OF CONTENTS

IMPORTANT NOTICE:

The information in this document is subject to change without notice and should not be construed as a commitment by Sinch. Sinch assumes no responsibility for any errors that may appear in this document.

1 INTRODUCTION

The SMS 365 HTTP interface is intended for Enterprises connecting into our messaging gateway for:

- Mobile Terminated (MT) Standard/Premium Services
- Mobile Originated (MO) Standard/Premium Services

This HTTP interface requires Enterprises to develop an HTTP client (for sending MT messages to us), as well as an HTTP server (for receiving MO and MT message acknowledgments). Detailed explanations and samples are provided throughout the document. For additional information and support on the HTTP interface, contact your Account representative.

2 GETTING STARTED

The following is provided by us and is required in order to use the HTTP interface:

- valid username and password
- account specific URL

Note: Please contact your account manager to obtain the above information.

- *Gateway URL for MTs*: Depending on their location clients may want to connect to data centers in Europe or in the USA.

Descripton	URL	IP address
Europe, VPN	http(s)://sms-pi.sapmobileservices.com/cmn/	178.248.228.11
Europe, non-VPN	http(s)://sms-pp.sapmobileservices.com/cmn/	178.248.228.151
USA, VPN	http(s)://sms-cp.sapmobileservices.com/cmn/	74.117.12.163
USA, non-VPN	http(s)://sms-ci.sapmobileservices.com/cmn/	74.117.12.34

- *The Gateway IP for MOs and Delivery Notifications:* **178.248.228.121.** Make sure these are registered with the firewall as all MOs and delivery notifications will come from those IP addresses.
- Depending on the configuration customers might receive MOs and delivery notifications from a different IP address. If that's the case the account manager will let you now. For instance:
  - Customers connecting to the sms-cp.sapmobileservices.com/mm address might receive MOs and Delivery Notifications from 74.117.12.163
  - Customers connecting to the sms-ci.sapmobileservices.com/mm address might receive MOs and Delivery Notifications from 74.117.12.34
- Gateway Ports: HTTP/**80**; HTTPS/**443**

In addition, it is required that you provide the following information to your account manager:

- Reply URL at Enterprise server (if applicable) for receiving HTTP MO messages
- Enterprise MT posting server IP address (for access to the Gateway)

3 MT HTTP INTERFACE

3.1 HTTP Request for Sending Messages

The Enterprise application can send SMS messages to the HTTP interface using the HTTP POST method. The HTTP request is composed of two sections: the *HTTP Header* and the *HTTP Body*. The format for sending messages to the HTTP interface is shown below in Table 1:.For a list of mandatory and optional parameters, see *Table 3:, page 10*, and *Table 4:, page 12*.

Note: Mandatory fields correlate to specific message types; for example, the **Binary** field cannot be used to send a text message.

Table 1: Example of an HTTP Request

Sample	
HTTP Header	POST /cmn/customer_account/customer_account.sms HTTP/1.1 HOST: sms-pp.sapmobileservices.com Authorization: Basic TW9iaWxlMzY1Ok0zNjU= Content-Length: 95
HTTP Body	Subject=Example [MSISDN] List=+44777123123,+44777123124,+44777123125 [MESSAGE] Text=A SAMPLE SMS to three mobile phones. [SETUP] MobileNotification=YES [END]

3.1.1 HTTP Header

HTTP requests are sent to the following path on the **sms-pi.sapmobileservices.com** server:
/cmn/customer_account/customer_account.sms

Each page is unique to one customer account. For example, the customer account for customer ABC is called *ABC_Singapore_HTTP* and the URL is:
http:// sms-pi.sapmobileservices.com/cmn/ABC_Singapore/ABC_Singapore.sms

Note: Account-specific URLs are created prior to testing and implementation and are not accessible using a Web browser.

Basic Authorization is supported as specified by RFC 2617. For individual customer validation, the **Authorization** field requires an encoded login and password, separated by a colon (:).

Table 2: Example of an HTTP Header

Sample
Login: login Password: password Authorization: Basic bG9naW46cGFzc3dvcmQ= (where bG9naW46cGFzc3dvcmQ= login:password encoded in base 64)

For details about HTTP Headers and other HTTP questions, please refer to RFC 2616.

3.1.2 HTTP Body

The mandatory parameters used to send a message are explained below in Table 3: page 10.
Parameters are separated from the value using the equal sign (=). No spaces are allowed between the parameter, the equal sign, and the value.

Table 3: Mandatory Parameters

Section	Parameter	Value
[MSISDN]	List	MSISDN in international format (E.164 number format) with a plus sign (+). If the same message needs to be sent to multiple mobile numbers, up to 100 MSISDNs can be listed and separated by a comma (,); more than 100 MSISDNs must be split into multiple requests. No space is allowed between the comma and the phone number.
[MESSAGE]	Text	Specifies the text that is sent to a mobile phone. This field is mandatory if you send a text message, and in this case, the Binary and Length fields should not be present. Six character sets are accepted: SAP Characters Set (SCS), Chinese char set in BIG5, Chinese char set in GB2312, Greek char set in Latin/Greek 8859-7 And Base 64 encoded UTF8. If the Text length is bigger than 160 characters, the messaging gateway splits the message and adds the appropriate UDH information. If the operator and handset support it, this information displays on the handset as a single long text message. Refer to samples 3.3.28-3.3.31. If you don't want this automatic split text feature to be enabled, the optional parameter SplitText shall be set to NO. Please refer to Table 4: below. Please see 3.1.3 for further details Few GSM extended characters are treated as 2 bytes hence bytes counting may be different from what you see if text contains any of them. Refer to *Appendix A* for the list of GSM extended characters.
[MESSAGE]	Binary	Only used when it is a Binary message or a Unicode characters (UCS2) message, represented in hexadecimal string. If a message with a length >140 bytes is sent, the platform will cut the message in multipart and generate the concatenation information. Each part of the message is one SMS and therefore message will be billed as multiple SMS. Maximum length for Binary messages (such as ringtones) is 1024 bytes (example: 2048 ASCII characters, no spaces).
[MESSAGE]	Length	Mandatory when Binary is present (Binary or UCS2 message). Represents the number of binary bytes before encoding (for example, the length of a hexadecimal string *4f60* is *2*.

Section	**Parameter**	**Value**
[SETUP]	OriginatorPort	Only used when it is Binary message, to specify the application port number. The following are a few Nokia Smart message port numbers: 1581 = Ringtone 1582 = Operator logo 1583 = Calling line identification (CLI) logo 158A = Picture Message 00E2 = vCard 00E4 = vCalendar For WAP push: OriginatorPort=23F0 Refer to section *0,* *Sending Binary* Messages*, page 18*.
[SETUP]	DestinationPort	Only used when it is Binary message, same as OriginatorPort when it is Nokia Smart message. For WAP push: DestinationPort=0B84
[SETUP]	DCS	Possible values are 8b, BIG5, UTF8, GB2312, 8859-7, BIG5 and UCS2. Specifies how the text of the short message is encoded. Mandatory when sending Chinese or other international characters, such as Greek, Arabian, Thai, or Korean. When nothing is specified and the message contains text, the value is automatically set to SAP Character Set. When the message contains binary data, this field is automatically set to 8b (representing 8 bits binary data). Refer to samples 3.3.2-3.3.5 and 3.3.18-3.3.21.

Table 4: below lists the optional parameters for submitting a message (if a field is not present, a default value is used). Optional parameters may be product or operator-specific. Please contact your account manager or pre-sales contact should if you have questions.

Table 4: Optional Parameters

Section	Parameter	Value
	Subject	Customer-provided reference text for tracking purposes is not sent to the mobile recipient. This field is returned with each acknowledgment and notification coming from the platform. The length should not exceed 255 characters. This field can be useful to set your own unique ID which will be returned in notifications and will ease tracking of messages statuses.
[SETUP]	AckType	This field indicates the level of notification the platform will send back to the customer. Possible values include: Order: When this option is set, the platform only acknowledges receipt of the message by giving an order identifier and the number of messages generated in the HTTP response. Message: In addition to the Order notification level, this option requests for SMS-C or Handset acknowledgments. This is the default value. Refer to sample 3.3.26.
[SETUP]	AckReplyAddress	Customer-provided URL for receiving message acknowledgment. If no URL is set no delivery notifications will be sent however status of messages will be visible in the system as well as in the customer extranet. If customers would like to request for optional parameters to be received in the delivery notifications an additional string (‘?SAPF=XXX’’ shall be appended to the URL. See 0 for details on how to request optional parameters. The URL must not be longer than 99 characters.
[SETUP]	MobileNotification	Only used when a customer requests to receive Mobile Notifications. A *Mobile Notification* is an operator-dependent feature. Possible values include Yes or No. The default value is No. Refer to sample 3.3.26.
[SETUP]	OriginatingAddress	Used only when a customer would like to specify the OriginatingAddress (TPOA) value (or is advised to do so). It is an operator-dependent feature and may not be available for all destinations.

Section	Parameter	Value
[SETUP]	ValidityPeriod	Used only when the customer wants to specify the *Validity Period*. A Validity Period is assigned to each short message submitted to the platform, setting the maximum time that the short message is retained in the platform's network and/or in the network of the corresponding wireless operator. Setting specific short message Validity Period is important for many SMS-based applications. For example, a daily short message informing you of a stock quotation at the end of the day would have a 24-hour validity period. In this case, the information is not necessarily relevant if received more than 24 hours after being sent. The validity period defined when sending a HTTP request to the platform is independent of the maximum validity period allowed on each specific wireless network. 1 week is the maximum possible value, 5m is the smallest possible value. The validity period starts the moment the platform receives the HTTP request. Coding: ValidityPeriod=n[w|d|h|m] Where: *n=number of units* *w=week* *d=day* *h=hour* *m=minute* Only the two first characters of the parameter are taken into account, meaning that only one value can be set. Examples: ValidityPeriod=1w ➔ One week ValidityPeriod=3d ➔ Three days If this value is not specified by the customer, the platform uses the GSM default validity period setting: 48hrs.

[SETUP]	Class	Used to specify the message class. Message class is an operator-dependent feature.
		Class=0 Immediate display (flash)
		Class=1 Handset Specific (SAP Default)
		Class=2 SIM Specific
		Class=3 TE Specific
		The Class values are described in GSM TS 03.38. (3GPP 23.038)

Section	**Parameter**	**Value**
[SETUP]	OperatorId	Used to specify the destination operators for the message. If there is no operator ID specified, the platform determines the destination operator based on its global numbering plan. In order to specify an operator ID, please see the Operator List document for connected operators and their corresponding operator IDs. The platform network can only accept operator IDs from this list. Please refer to sample 3.3.23 to see how to use the OperatorId parameter.
[SETUP]	PID	By default, the PID set by the Gateway is equal to 0x00 (0 in Hexadecimal format). When a special value is needed, it is possible to define this value inside the request by giving the hexadecimal code of the PID desired. This feature is only supported by a few operators so please consult your account manager if you are interested in using it. For example: [SETUP] PID=2A Note: This information is independent from all other fields given in input.
[SETUP]	UDH	Alternative method of sending Binary messages without using a port number. UDH and UDHL must be used together. There are two different ways to use these two parameters. For more information, see section *0, Sending Binary* Messages*, page 18*.
[SETUP]	UDHL	See section *0,* *Sending Binary* Messages*, page 18*.
[SETUP]	SESSION_ID	Optional parameter for session tracking or other purposes. When supported by an operator, you may receive information in the SESSION_ID field of an incoming MO request. Upon receiving such information, you are expected to post it back into the SESSION_ID field of the MT reply. Enterprises may need to send billing or keyword information in this field. Formatting for billing is currently dependent on the billing platform. This field is also used for keyword caching (for details, see *section3.1.2.1, Keyword* Caching*, page 16*). Refer to sample 3.3.25.

Section	Parameter	Value
[SETUP]	SplitText	If you don't want long text messages (> 140 Bytes on the handset) to be splitted by the messaging gateway, you can add this field and set it to NO. The messaging gateway will then refuse messages longer than 160 characters. Note: no and NO are accepted. Note2: default value for SplitText is YES. Note3: Additionally the platform can configure your account to split messages longer than 140B and adding X/X at the beginning of each part. For example message1 with "1/2 hello world…", and message2 with "2/2 end of message". Please check with your account manager.
[END]		

3.1.2.1 Keyword Caching

The platform can provide a keyword caching functionality. When active, a keyword can be passed using the <cache> tag in the **SESSION_ID** field (i.e., *SESSION_ID=<cache>keyword</cache>*). An MT message is sent to an MSISDN using the **<cache>** tag. For a specified frequency, any MO coming from this MSISDN and addressed to a specified short code will have this keyword activated. This feature is currently available in Germany only and on a limited list of short codes and operators, please contact your account manager should you have any questions.

3.1.3 Sending Text Messages

3.1.3.1 Message length

Most networks' A2P gateways support 140Bytes SMS messages. The Default GSM Characters Set (DGCS) is 7bits encoded, meaning it can support up to 160 characters in one SMS. However some characters, as the € sign, use the space of two characters, the escape character plus a character from the main table.

When we refer to 160 characters as the maximum length we refer to what is actually received by the handset, we do not refer to what is submitted by the Content Provider. For example we ask to send "<LF>" for sending a Line Feed, which is 4 characters, but it will be received as only one character by the handset. The opposite example is the pipe character "|", which can be submitted as only one character by the Content Provider however it will take the space of 2 characters in the handset.
Please refer to the default Char Set table in Appendix A part I, and the extended CharSet in part II.

Note: In this document when we refer to the message length, whether in octets/bytes or in number of characters, we mean the length as what actually reaches the phone.

3.1.3.2 Message Encoding

The **TEXT** field is used when sending text. Possible values and corresponding DCS values are listed below:

- If text is coded in SAP Character Set format, the maximum permitted length for this field is 160 characters. If one of the characters is a GSM-extended character (such as € ^ { } [ ] ~ \ |), it will be counted as two characters. <LF> is recommended as a line feed character rather than <CR>, as most handsets are incompatible with <CR>. The list of permitted characters is the GSM Default Alphabet in ***Appendix A: SAP Character Set**, page 73.*

  No DCS field is required, the default setting, SAP Character Set, is used.

  Please refer to sample 3.3.2.

- If text is coded in Latin/Greek (ISO 8859-7) format, the maximum permitted length for this field is 160 characters (140 bytes). This character set is mainly used for sending Greek characters.

  The DCS value should be specified as DCS=8859-7 in this case. ***Appendix B: Greek Characters in Latin*/Greek 8859-7***, page 79* lists supported Greek characters.

  The list of permitted characters from ISO 8859-1 is the GSM Default Alphabet in ***Appendix A: SAP Character Set**, page 73*.

- If text is coded in BIG5 format for Traditional Chinese Characters, the maximum permitted length is 70 characters. The specific markers <CR> and <LF> are also available.

  In this case, the DCS field is specified as DCS=BIG5. Please refer to sample 3.3.4.

- If text is coded in GB2312 format for Simplified Chinese Characters, the maximum permitted length is 70 characters. The specific markers <CR> and <LF> are also available.

  In this case the DCS field is specified as DCS=GB2312. Please refer to sample 3.3.3.

- The UTF-8 text field can contain any character that is in the UCS2 character set. The maximum permitted length for this field is 140 octets. All UTF8 encoded messages must be base64 encoded. The carriage return and line feed should not be escaped.

  In this case, the DCS field must be specified as DCS=UTF8.

  All GSM characters are accepted, as noted in ***Appendix A: SAP Character Set****, page 73*. Please refer to sample 3.3.27.

  If all characters are part of the Default GSM Characters Set (DGCS) the maximum permitted length is 160 characters

  If any of the character is not in the GSM Default Alphabet, THE PLATFORM will encode the message in UCS2. Maximum permitted length is 70 characters.

  Note: The list in ***Appendix A: SAP Character Set****, page 73* corresponds to what the messaging gateway supports. However, because all operators and handsets each have limitations, some characters in certain cases may not display correctly on a handset.

3.1.3.3 Sending as UCS2 Binary

The **Binary** field is used for sending UCS2 encoded messages (Chinese, Arabic, cyrilic or other International language characters).

The maximum permitted length for this field is 70 Characters (one character is using 2 octets in UCS2 encoding). The length must be specified as the number of bytes in the Binary field. In this case, the DCS field is specified as DCS=UCS2.

Please refer to samples 3.3.18-3.3.21.

3.1.4 Sending Binary Messages

Binary messages can be sent using two different methods: using port information or using UDH information. This section explains in detail how to send binary messages.

3.1.4.1 Using Port Information (Recommended)

This is the best way to send a binary message if the Enterprise is not able to generate UDH for binary messages.

In order to use **OriginatorPort** and **DestinationPort**, binary messages must be sent in one request, even if that request is longer than 140 bytes. UDH information is added by the messaging gateway according to port information and the length of the message. If the binary content is longer than 140 bytes, the message is split into multiple parts, which are known as concatenated binary messages.

- Single Binary message sending—refer to sample 3.3.5
- Long Binary message sending—refer to sample 3.3.6
- WAP push message sending—refer to sample 3.3.10

3.1.4.2 Using Complete UDH Information

This is the best way to send preformatted binary messages to SAP (if the Enterprise is able to split long binary messages and add UDH accordingly). For example, when sending a three-part ringtone message, the Enterprise must send three requests with each part in one request.

Table 5: below shows how to construct UDH and UDHL for single and long binary messages.

Table 5: Using Complete UDH Information

Message Type	**UDH**	**UDHL**
single binary message	0504XXXXYYYY	6
long binary message	0504XXXXYYYY0003AABBCC	11

Where:
XXXX=DestinationPort
YYYY=OriginatorPort
AA=Remains constant for every message that makes up the concatenated message (2 octets from 00 to FF) unique in the system.
BB=Maximum number of short messages in concatenated, value remains the same for all segments in the concatenated message
CC=Sequence number of current short message incremented by one for each following sequence.

- Single Binary message sending—please refer to sample 3.3.7
- Long Binary message sending—please refer to sample 3.3.8
- WAP push message sending—please refer to sample 3.3.11

3.1.4.3 Using Generic UDH Information (Recommended)

This is a very simple way to send binary message using a generic UDH and UDHL. Binary messages must be sent in one request, even if that request is longer than 140 bytes.
Since messages are sent in one request, UDH and UDHL are always using the same generic value, regardless of whether your message is a long concatenated one or a short one.

Table 6: Single and Binary Messages Using UDH and UDHL

Message Type	**UDH**	**UDHL**
single binary message	0504XXXXYYYY	6
long binary message	0504XXXXYYYY	6

Where:
XXXX=DestinationPort
YYYY=OriginatorPort

The messaging gateway adds the reference number, maximum number and sequence number for each concatenated message to generate a real UDH.

- Single Binary message sending—refer to sample 3.3.7
- Long Binary message sending—refer to sample 3.3.8
- WAP push message sending—refer to sample 3.3.11

3.1.5 Optimizing Throughput

For customers running services requiring high throughput, we recommend the following:

- Submit multiple concurrent POST requests to messaging platform.
- For bulk send using our **List** parameter (see section 3.1.2), send the maximum number of MSISDNs in each submission (100).
- User HTTP 1.1 persistent mode. The HTTP gateway responds to HTTP POST requests with a ‘connection: close’ in the header, however it does not close the TCP connection for 3 seconds, therefore clients can submit POST requests within the same TCP connection as long as there is at least one such request every 3 seconds. After 3 seconds of inactivity HTTP server will send a TCP command to the client to inform it of the closure of the connection. If a client attempts to submit a message using the same TCP connection after 3 seconds of inactivity then HTTP server will respond with an error.

3.1.6 Requesting for optional parameters in the delivery notifications

The service provides optional parameters which returns additional information on the status of a message. The parameters are available on demand.

3.2 HTTP Response for Sending a Message

In response to an HTTP request, the server sends an HTTP response as a Gateway acceptance acknowledgment. If the data sent has been processed successfully by the Gateway, the HTTP response gives one order ID for each HTTP request.

The following is an example of a positive HTTP response:

Table 7: Example Positive HTTP Response

Sample

HTTP/1.1 200 OK
Content-Length: 89
Content-Type: text/html
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<HTML>
<HEAD>
</HEAD>
<BODY>
#Message Receive correctly
ORDERID=1436496749
</BODY>

More than one order ID separated by a comma (,) are generated if (and only if) the message is split over several short messages.

In the event that an error occurs, an error code and a message explaining the reason for the error is returned. Table 8: is a list of negative HTTP responses. This message is generated when the application has a problem with the message submission, in most case it occurs when the application has detected an incorrect parameter or an invalid field inside the body of the message.

Table 8: Example Negative HTTP Response

Sample
HTTP/1.1 400 Bad Request Content-Length: 59 Content-Type: text/html <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN"> <HTML> <HEAD> </HEAD> <BODY> 6100 Error processing message </BODY>

3.2.1 HTTP Interface Response Error Codes

Table 9: Response Error Codes

Error	**Description**	**What Next?**
4450	Message rejected due to MO exceeded threshold limit set for Account.	Check the MO threshold limit for the Account.
4451	Message rejected due to MO exceeded threshold limit set for the same MSISDN.	Check the MO threshold limit for the same MSISDN.
5001	Server currently unavailable to process your message; please try later.	Raise a trouble ticket to Service Desk.
5002	Server currently unavailable to process your message; please try later.	Raise a trouble ticket to Service Desk
5003	Server currently unavailable to process your message; please try later.	Raise a trouble ticket to Service Desk
5004	Server currently unavailable to process your message; please try later.	Raise a trouble ticket to Service Desk
5005	Server currently unavailable to process your message; please try later.	Raise a trouble ticket to Service Desk
5006	Server currently unavailable to process your message; please try later.	Raise a trouble ticket to Service Desk.
600F	Authentication failure	Raise a trouble ticket to Service Desk
6010	Bad format for field DCS in section SETUP.	Check syntax of DCS field.
6011	Bad format for field PID in section SETUP.	Check syntax of PID field.
6012	Bad format for field MobileAck in section SETUP.	Check syntax of MobileAck field.
6013	Bad format for field AckType in section SETUP.	Check syntax of AckType field.
6017	Bad format or missing mandatory value for field Text in section MESSAGE	Text or Binary fields must contain a value.
6018	Bad format or missing mandatory value for field List in section MSISDN	List field must contain a value.

Error	Description	What Next?
6019	Bad format for field List in section MSISDN.	Check syntax of MSISDN field (comma-separated).
601A	Bad format for field Class in section SETUP.	Check syntax of Class field.
6024	Bad format for field ValidityPeriod in section SETUP.	Check syntax of ValidityPeriod field.
6026	Bad format or missing mandatory value for field DestinationPort in section SETUP	The DestinationPort field is empty.
6028	Bad format or missing mandatory value for field Length in section MESSAGE.	Length and Binary fields must exist.
6029	Bad format or missing mandatory value for field Binary in section MESSAGE.	binary message must be after parameter Binary
602A	The message contained an invalid character.	Check the message characters against the list in ***Appendix A: SAP Character Set****, page 73.*
602B	The message contained an invalid binary string length.	Compare the length of the Binary and Length fields.
602C	The message contained a version value error.	Raise a trouble ticket to Service Desk
602D	The message contained a reply URL too long.	Check the URL length.
6030	Error when processing the message.	Raise a trouble ticket to Service Desk
6037	Invalid UDH value	Check the UDH field value.
6038	Wrong operator	The operatorid you set is not valid.
6050	Service unavailable	Temporary failure, please retry again later.
6100	Error when processing the message.	Check message syntax; check the accuracy of the character set used.
6101	Error when processing the message.	Check message syntax; check the accuracy of the character set used.
8000	Server currently unavailable to process your message; please try later.	Raise a trouble ticket to Service Desk

3.3 Sample MT Messages

3.3.1 Text: SCS Character Set with Line Feed Character

```
Subject=SCS Char set with line feed character

[MSISDN]
List=+6599990000

[MESSAGE]
Text=This is a text message. <LF> Courtesy of SAP.
[END]
```

3.3.2 Text: Greek Character Set with Line Feed Character

```
Subject= Greek character set with line feed character

[MSISDN]
List=+30123456789

[MESSAGE]
Text=Hello YOU'VE GOT SMS! Σ Υ Γ Χ Α Ρ Η Τ Ι Κ Ε Δ Ν Ε Ο Θ Λ Β Μ Ω Α Π Ε Ο Λ Φ  <LF> WWW.WEBSITE.GR

[SETUP]
DCS=8859-7
[END]
```

3.3.3 Text: Chinese Character Set in GB2312

```
Subject= Chinese Character set in GB2312

[MSISDN]
List=+6599990000

[MESSAGE]
Text=你好吗？ <LF> SAP Mobile Services

[SETUP]
DCS=GB2312
[END]
```

3.3.4 Text: Chinese Character Set in BIG5

```
Subject= Chinese Character set in BIG5

[MSISDN]
List=+6599990000

[MESSAGE]
Text=你好吗？ <LF> SAP Mobile Services

[SETUP]
DCS=BIG5
[END]
```

3.3.5 Binary: Operator Logo Message Using Ports (Single)

```
Subject= Operator Logo Message

[MSISDN]
List=+6599990000

[MESSAGE]
Binary=02F81000480E010000000000000000000000004140000000400000004040000000041C00000404000000067F01
D8E794E4939206FF812514551490520F
Length=133

[SETUP]
OriginatorPort=1582
DestinationPort=1582
[END]
```

3.3.6 Binary: Ringtone Message Using Ports (Long, Concatenated)

Long ringtone messages (> 140 bytes) have to be submitted as one message. The messaging gateway splits it and adds the appropriate UDH and UDHL using the **DestinationPort** and **OriginatorPort** information.

```
Subject= Ringtone Message
[MSISDN]
List=+60199990000

[MESSAGE]
Binary=024A3A51214929280400FD1AA2A82CC2AC2652A82CC2AC2652A82CC2AC22822C4984164156164288B0
8952618A2A82CC2AC265498628AA0B30AB09952618618A2A82CC2AC22822C4984166156164288B08952610A2A
C2252A82CC3042A82CC3082C82A826C2A422826C2A826822849C6288908A08B0C108A08B0B20AB0B30AB0B2
0C30AA0994AA0B30C10AA0B30C20B20AA09B0A908B09B0AA09A08A12718A22422822C30422C22C2CC2AC2C
C498628AA09B08B0890000
Length=183

[SETUP]
DestinationPort=1581
OriginatorPort=1581
[END]
```

3.3.7 Binary: Ringtone Message Using UDH (Single)

In addition to using **DestinationPort** and **OriginatorPort**, Customer can also use UDH and UDHL when sending binary messages.

```
Subject=Ringtone Message using UDH and UDHL
[MSISDN]
List=+60169990000

[MESSAGE]
Binary=024A3A54C4C0D8C8DC04006CD8A49561381569861C6288C08B49B0AB0B409B4AB08A5269862069A618
6986184586185485584E055A618718A23022D26C2AC2D026D2AC22822C49C61A4289A08B12718698718A22500
0
Length=86

[SETUP]
UDH=050415811581
UDHL=6
OriginatingAddress=8088
[END]
```

3.3.8 Binary: Ringtone Message Using Complete UDH (Long, Concatenated)

Concatenated messages have to be sent individually with complete UDH.

```
Subject=Long Ringtone Message using UDH and UDHL part 1
[MSISDN]
List=+60169990000

[MESSAGE]
Binary=024A3A6D3995D1A195C9B185B991CC0400D8D513710A18690B18710A1A91C7909AC8B1469E42671C42
86A471E426B22D21051C41A91881081A710A18190793710B98710B98710B9A91C7909AC8B146A242E7106A471
E426B22851C710A1A91879099A41819079C92C8BC84C9A8858AA09908A1471C42E6A4722428B22851C710
Length=128

[SETUP]
UDH=0504158115810003120201
UDHL=11
OriginatingAddress=8088
[END]
```

```
Subject=Long Ringtone Message using UDH and UDHL part 2
[MSISDN]
List=+60169990000

[MESSAGE]
Binary=B9A41841A510B13710A1861761891A61C41A510B18710B97710B93415917618710A184174182907800
Length=41

[SETUP]
UDH=0504158115810003120202
UDHL=11
OriginatingAddress=8088
[END]
```

3.3.9 Binary: Ringtone Message Using Generic UDH (Long, Concatenated)

Long ringtone messages must be sent as one message, and the UDH and UDHL are the same as the single message. The messaging gateway adds the reference number, maximum number, and sequence number to each split message in order to generate a real UDH, such as *0B0504158115810003120201* and *0504158115810003120202*.

```
Subject=Long Ringtone Message using UDH and UDHL
[MSISDN]
List=+60169990000

[MESSAGE]
Binary=024A3A51214929280400FD1AA2A82CC2AC2652A82CC2AC2652A82CC2AC22822C4984164156164288B0
8952618A2A82CC2AC265498628AA0B30AB09952618618A2A82CC2AC22822C4984166156164288B08952610A2A
C2252A82CC3042A82CC3082C82A826C2A422826C2A826822849C6288908A08B0C108A08B0B20AB0B30AB0B2
0C30AA0994AA0B30C10AA0B30C20B20AA09B0A908B09B0AA09A08A12718A22422822C30422C22C2CC2AC2C
C498628AA09B08B0890000
Length=183

[SETUP]
UDH=050415811581
UDHL=6
OriginatingAddress=8088
[END]
```

3.3.10 Binary: WAP Push Message

```
Subject=WAP Push Message
[MSISDN]
List=+60169990000

[MESSAGE]
Binary=000601AE02056A0045C60C0336312E382E3231352E34342F7A696E672F7A696E676C6574563F7461736B
3D7626703D353234390001034D4D532052656365697665564000101
Length=69

[SETUP]
OriginatorPort=23F0
DestinationPort=0B84
[END]
```

3.3.11 Binary: WAP Push Message Using UDH and UDHL

```
Subject=WAP Push Message
[MSISDN]
List=+60169990000

[MESSAGE]
Binary=000601AE02056A0045C60C0336312E382E3231352E34342F7A696E672F7A696E676C6574563F7461736B
3D7626703D353234390001034D4D532052656365697665564000101
Length=69

[SETUP]
UDH=05040B8423F0
UDHL=6
[END]
```

3.3.12 Binary: Ringtone EMS (Single)

In order for the messaging gateway to successfully deliver the message, insert the EMS information into the **UDH** field, and populate the **Binary** field with a random value.

```
Subject=Ringtone EMS
[MSISDN]
List=+123456789

[MESSAGE]
Binary=3030
Length=2
[SETUP]
UDH=0C7C004D454C4F44593A2A3461332A3372336133723361337233613372332A3461332A337233613372332A3461332A
3372336133723361337233 2A3461332A3372336133723366336533 66312A3461332A337233613372332A3461332A
3372336133723361337233 2A3461332A33723361337233653372336433 72332A3361330D0A
UDHL=126
[END]
```

3.3.13 Binary: Picture EMS (Single)

In order for the messaging gateway to successfully deliver the message, insert the EMS information into the **UDH** field, and populate the **Binary** field with a random value.

```
Subject=Picture EMS
[MSISDN]
List=+33xxxx

[MESSAGE]
Binary=6161
Length=2

[SETUP]
UDH=1081020000000000000000000F8000003060000040100000400800005 00C00FFE817FFFF807C7FFFC0F03F
FFD0E03FFFF0C07FFFE180CFFFF1010FFFE3020FFFE2040FC064080C3FE41013FFC2202FFFC3E04FFFC1E08
FFF80F10FFF807E0FC3807E0E3F803F19FF803FE7FFC061E80040807000400070008001C00000000000000000
UDHL=131
[END]
```

3.3.14 Binary: Picture + Logo EMS

Two requests to the messaging gateway are required as follows:

```
Subject=Picture + logo EMS
[MSISDN]
List=+123456789

[MESSAGE]
Binary=6161
Length=2

[SETUP]
UDH=1081020000000000000000000F800000306000004010000040080000500C00FFE817FFFF807C7F
FFC0F03FFFD0E03FFFF0C07FFFE180CFFFF1010FFFE3020FFFE2040FC064080C3FE41013FFC2202FFFC3
E04FFFC1E08FFF80F10FFF807E0FC3807E0E3F803F19FF803FE7FFC061E80040807000400070008001C
00000000000000000
UDHL=131
[END]
```

```
Subject=Picture + logo EMS
[MSISDN]
List=+123456789

[MESSAGE]
Binary=4141
Length=2

[SETUP]
UDH=0C8002424547494E3A494D454C4F44590D0A56455253494F4E3A312E300D0A464F524D41543A
434C415353312E300D0A4D454C4F44593A2A336633663366332363312364332364332364336331723
366336633663323633323663323663323663366332A342363332363332363332A332361310D0A454E
443A494D454C4F44590D0A0003550202
UDHL=135
[END]
```

3.3.15 Binary: Chinese EMS

```
Subject=Chinese EMS

[MSISDN]
List=+33603258257


[MESSAGE]
Binary=4E2D6587
Length=4

[SETUP]
DCS=UCS2
UDH=0C7C004D454C4F44593A2A3461332A33372336133723361337 2332A3461332A3372336133723
32A3461332A3372336133723361337 2332A3461332A3372336133723366336533663 12A3461332A3
37233613372332A3461332A33723361337233613372332A3461332A33723361337233653372336433
72332A3361330D0A
UDHL=126
[END]
```

3.3.16 Binary: Long EMS Message Using Generic UDH

```
Subject=Very_Long_EMS

[MSISDN]
List=+123456789

[MESSAGE]
Binary=300000115465737420746F204D6F62696C6533363502010000481C01000000000000000000
000000000000000000000000000000000000155550000000000000119B68000000000000125960000
0000000001F56380000000000001F34600000000000001A48B8000000000000122510000000000000
114D78000000000000121130000000000000018DA0800000000000018D5800000000000001BC84800
0000000000183D800000000000001484F80000000000001DA2C000000000000016BBA80000000000
001F67900000000000001FFFF80000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000
00
Length=280

[SETUP]
udh=0504158A0000
udhl=6
mobilenotification=Yes
[END]
```

3.3.17 Binary: Long EMS Message Using Complete UDH

Concatenated messages must be sent individually with a complete UDH.
The example below displays the same content as the previous example (3.3.16).

```
Subject=Very_Long_EMS_Part1

[MSISDN]
List=+123456789
[MESSAGE]
Binary=300000115465737420746F204D6F62696C6533363502010000481C01000000000000000000
000000000000000000000000000000000000015555000000000000119B68000000000000125960000
0000000001F56380000000000001F34600000000000001A48B8000000000000122510000000000000
114D7800000000000012
Length=128

[SETUP]
UDH=0504158A00000003010301
UDHL=11
[END]
```

```
Subject=Very_Long_EMS_Part2

[MSISDN]
List=+123456789

[MESSAGE]
Binary=113000000000000018DA0800000000000018D5800000000000001BC848000000000000183D
800000000000001484F80000000000001DA2C000000000000016BBA80000000000001F6790000000
0000001FFFF8000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000
Length=128

[SETUP]
UDH=0504158A00000003010302
UDHL=11
[END]
```

```
Subject=Very_Long_EMS_Part3

[MSISDN]
List=+123456789

[MESSAGE]
Binary=000000000000000000000000000000000000000000000000
Length=24

[SETUP]
UDH=0504158A00000003010303
UDHL=11
[END]
```

3.3.18 UCS2: Chinese Message with TPOA

```
Subject=Chinese Message with TPOA
[MSISDN]
List=+60169990000

[MESSAGE]
Binary=7b2c4e09573a003a00200028003300290020706b6eda00200020002000280031003000290020
601d7ef45c0f6cb300280033661f63a88350ff010029
Length=60

[SETUP]
DCS=UCS2
OriginatingAddress=36200
[END]
```

3.3.19 UCS2: Greek Character Set Message

This message is displayed as **ABCαβ** on a handset that supports Greek characters.

```
Subject=Greek character set message

[MSISDN]
List=+60169990000

[MESSAGE]
Binary=00410042004303B103B2
Length=10

[Setup]
DCS=UCS2
[END]
```

3.3.20 UCS2: Russian Character Set Message

```
[MSISDN]
List=+33661702896
[MESSAGE]
Binary=041D0430043F04380448043804420435002C0020043F043E04360430043B04430439044104
420430002C002004410432043E043500200441043E043E043104490435043D0438043500200437043
404350441044C002E00200414043B0438043D043000200441043E043E043104490435043D0438044
F0020043D043500200434043E
Length=166
[SETUP]
DCS=UCS2
[END]
```

3.3.21 UCS2: Arabic Character Set Message

```
[MSISDN]

List=+966123456789

[MESSAGE]

Binary=062F06480631064A00200623062806370627064400200623064806310648062806270648004
30041004C00430049004F064806270644064506320 64A062F00200645063900200634064806 2A0627
064A06450020064800410052005406450646003100350035 0631064A06270644002F0020063406470
631003000310032003700330034003500300035

Length=94

[SETUP]

DCS=UCS2
[END]
```

3.3.22 UCS2: emoji

EMOJI / emoticons characters are supported as part of the UNICODE standard.

- Ensure the data coding schema (DCS) used is UCS2, when encoding the messages and that you understand the constraints that UCS2 encoding introduces to the message and message length.
- See the EMOJI Wiki page for more details: https://en.wikipedia.org/wiki/Emoji

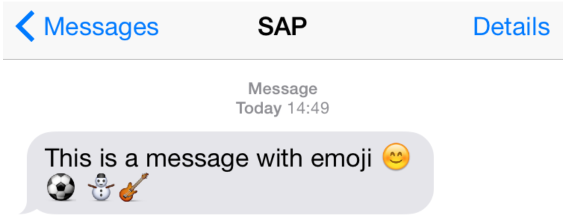
Note: This applies only to MT messages. Additionally, these characters are only be supported on select Operators that support UCS2.

```
List=+353123456789
[MESSAGE]
Binary=0054006800690073002000690073002000610020006D00650073007300610067006500200 0
770069007400680020 0065006D006F006A00690020D83DDE0A002026BD002026C4D83CDFB8
length=27

[SETUP]
OriginatingAddress=SAP
DCS=UCS2

[END]
```

On an iPhone it will display the following:



Message
Today 14:49

This is a message with emoji 😊 ⚽ ⛄🎸

3.3.23 General: Message with OperatorID

```
Subject=Message with OperatorId

[MSISDN]
List=+6598990000

[MESSAGE]
Binary=00410042004303B103B2
Length=10

[SETUP]
DCS=UCS2
OperatorId=61
[END]
```

3.3.24 General: Message with SMSC ACK request

```
Subject=ABC_60169990000_8080025

[MSISDN]
List=+60169990000

[MESSAGE]
Binary=003C0048006F0074006D00610069006C003E0020005A0069007A006100770061003A0066007
7003A0020004C006900630065006E00630065002000
Length=59

[SETUP]
DCS=UCS2
OriginatingAddress=8080025
AckType=MESSAGE
AckReplyAddress=http://example.com/ABC/MTAckReceive.aspx
[END]
```

3.3.25 General: Message with SESSION_ID

```
Subject=ABC_60169990000_8080025

[MSISDN]
List=+60169990000

[MESSAGE]
Binary=003C0048006F0074006D00610069006C003E0020005A0069007A006100770061003A0066007
7003A0020004C006900630065006E00630065002000
Length=59

[SETUP]
DCS=UCS2
OriginatingAddress=8080025
SESSION_ID=<cache>just for you</cache>
[END]
```

3.3.26 General: Message with Mobile Notification Request

```
Subject=Message with Mobile Notification request

[MSISDN]
List=+33619990000

[MESSAGE]
Text=Testing SMS from SAP. Have a nice day ahead!

[SETUP]
MobileNotification=Yes
AckType=MESSAGE
AckReplyAddress=http://example.com/api_php/MBW_DIR/notif.php
OriginatingAddress=89970020

[END]
```

3.3.27 UTF-8 Message

“test message” in base64 encoding

```
Subject=UTF-8 Message

[MSISDN]
List=+88123456789

[MESSAGE]
Text=dGVzdCBtZXNzYWdl

[SETUP]
DCS=UTF8
AckReplyAddress=http://example.com/SAP_client/SAP_client.cgi
[END]
```

3.3.28 Long Message: SCS Character Set

When submitting messages, if the length exceeds 160 GSM characters, the messaging gateway splits the message into multiple parts and adds the reference number, maximum number, and sequence number to each split message in order to generate the UDH. If the handset and the operator both support long text messages, this message displays as a single message on the user handset. The example below generates two messages, and two order IDs are received.

```
Subject=Text 161 characters

[MSISDN]
List=+1234567890

[MESSAGE]
Text=0123456789012345678901234567890123456789012345678901234567890123456789012345
678901234567890123456789012345678901234567890123456789012345678901234567890123456
7161

[END]
```

If you don’t want to get the SMS automatically splitted, the optional parameter SplitText shall be set to NO.

3.3.29 Long Message: BIG5

This example generates eight messages that are displayed as one message on the handset (assuming both the operator and the handset support it).

```
Subject= Long Message BIG5

[MSISDN]
List=+88123456789

[MESSAGE]
Text=你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services

[SETUP]
DCS=BIG5
[END]
```

The SplitText is by default set to yes.

3.3.30 Long Message: GB2312

This example generates six messages that are displayed as one message on the handset (assuming both the operator and the handset support it).

```
Subject=Long message GB2312

[MSISDN]
List=+88123456789

[MESSAGE]
Text=你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile Services你好吗？<LF> SAP Mobile
Services

[SETUP]
DCS=GB2312
[END]
```

3.3.31 Long Message: UTF8

This example generates two messages that are displayed as one unique message on the handset (assuming both the operator and the handset support it).

```
Subject=UTF-8 Message

[MSISDN]
List=+88123456789

[MESSAGE]
Text=MDEyMzQ1Njc4OTAxMjM0NTY3ODkwMTIzNDU2Nzg5MDEyMzQ1Njc4OTAxMjM0NTY3ODkwMTIzN
DU2Nzg5MDEyMzQ1Njc4OTAxMjM0NTY3ODkwMTIzNDU2Nzg5MDEyMzQ1Njc4OTAxMjM0NTY3ODkwMT
IzNDU2Nzg5MDEyMzQ1Njc4OTAxMjM0NTY3ODkwMTIzNDU2Nzg5MDEyMzQ1Njc4OTAxMjM0NTY3ODk

[SETUP]
DCS=UTF8
MobileNotification=yes
AckReplyAddress=http://example.com/SAPClient/SAPClient.cgi
[END]
```

4 DELIVERY NOTIFICATIONS

Delivery Notifications are message acknowledgments sent from the network to customers. When delivering messages through our network, customers can request to receive the following three levels of acknowledgment: *SAP Ack*, *SMS-C Ack*, and *Handset Ack*. For unsuccessful delivery attempts, the platform returns a negative acknowledgment (*Nack*) outlining the reason for the failure. Message acknowledgments are sent using the HTTP “GET” method to an Enterprise specified URL. In the case the messaging gateway doesn’t deliver the notifications successfully, it will retry up to 9 times, every 30 minutes.

Note 1: Enterprises must provide this notification URL to the platform in the MT request.

Note 2: Enterprises willing to receive DRs using HTTPS need to request the feature to be enabled to their account representative.

```
[SETUP]
AckReplyAddress=http://customerurl.com/customer.cgi
```

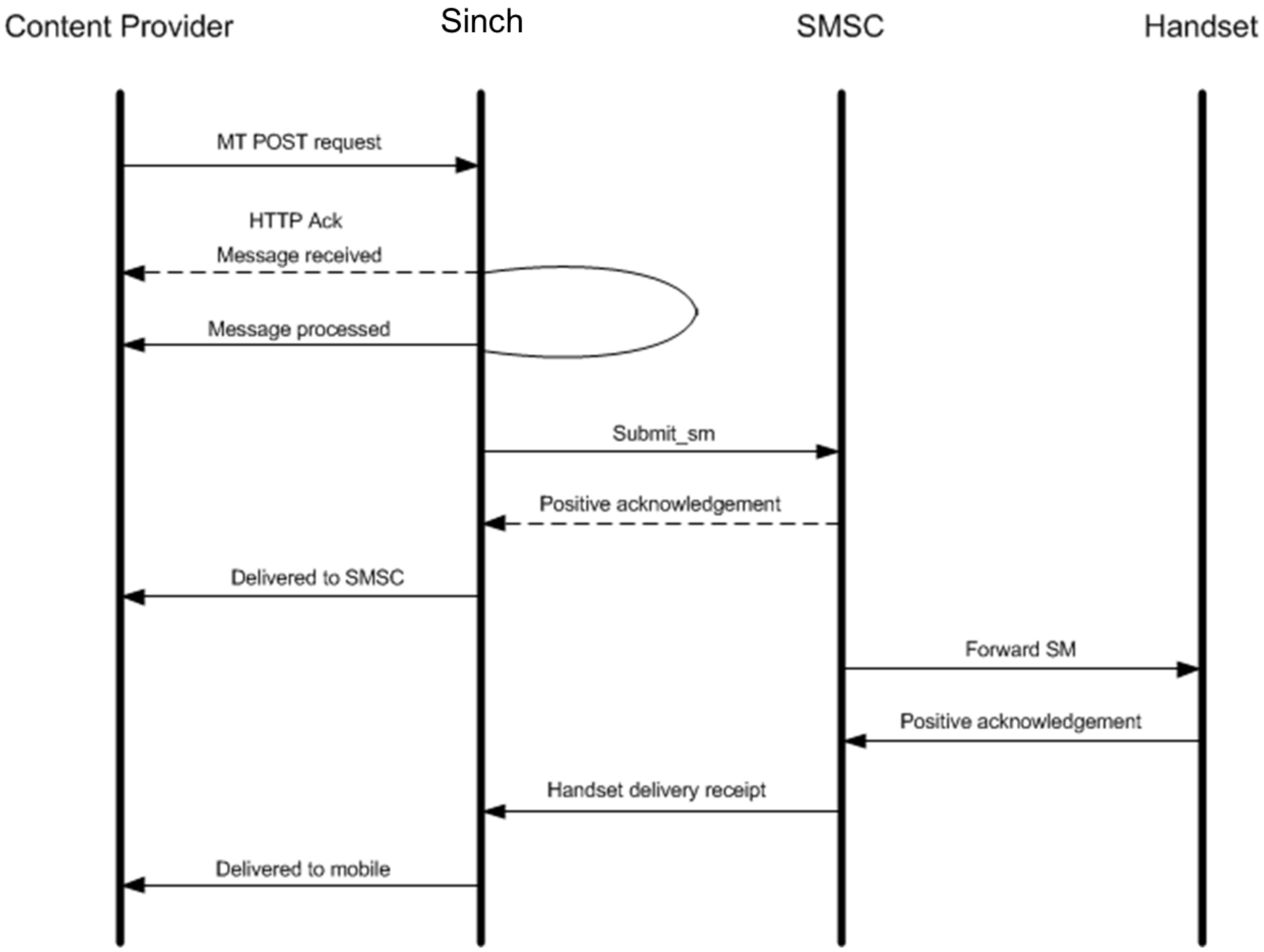


Figure a: Delivery Notification Flow

Table 10 demonstrates what needs to be included in the MT request and what kind of acknowledgment are sent back to Enterprises, provided the *Notification URL* is specified using *AckReplyAddress*.

Table 10: MT Request Contents

ACK Type	**Request Option**	**Notification Elements**	**Notes**
SAP Ack	AckType=ORDER or empty	CUSTOMERID>0 ORDERID>0 STATUS=text NBMESSAGE>0 SUBJECT=text	*SAP Notification:* Message acceptance--always received. Gives the number of short messages created and the order ID for all.

		CUSTOMERID=0
		ORDERID=0
		STATUS=text
		SUBJECT=text
			Negative Ack:
			Error processing message. Same as HTTP response message; STATUS is important.

SMS-C Ack	AckType= MESSAGE	CUSTOMERID>0 ORDERID>0 MESSAGEID>0 STATUS=text NBMESSAGES>0 SUBJECT=text MSISDN=phone umber DATE=date of notified event TIME=time of notified event	*SMS-C Notification:* DATE & TIME—date and time of the SMS-C acknowledgment
Handset Ack	MobileNotification= YES AckType= MESSAGE	CUSTOMERID>0 ORDERID>0 MESSAGEID>0 STATUS=text NBMESSAGES>0 SUBJECT=text MOBILEACK=1 MSISDN=phone number DATE=date of notified event TIME=time of notified event	*Handset Notification:* DATE & TIME—date and time the handset received the short message

4.1 Delivery Notifications Returned by the HTTP Interface

The following table captures the Delivery Notifications (both positive and negative acknowledgments) returned by the platform.

Table 11: Delivery Notifications Returned by the HTTP Interface

Status, as displayed on the extranet	Notification Indicates	Message Sent to Customer	Customer Receives (Example)	Retriable Notification? (Y/N)
Temporary Positive				
Message pending delivery to SMSC	SAP Ack	Your message has been correctly processed. <NBROFMESS> message has been generated with order ID <ORDERID>.	Your message has been correctly processed. 1 message has been generated with order ID 1886867226.	N
Waiting Mobile Notif	SMS-C Ack, waiting for Handset Ack	Message <MESSID> of the order <ORDERID> at destination of <MSISDNNBR> has been sent at <TIME> (CET) on the <DATE>.	STATUS = Message 1 of the order 1887109085 at destination of +6512345678 has been sent at 09:20:31 (CET) on the 11-05-2005.	N
Final Positive				
Delivered to SMSC	SMS-C Ack, not waiting for Handset Ack	Message <MESSID> of the order <ORDERID> at destination of <MSISDNNBR> has been sent at <TIME> (CET) on the <DATE>.	STATUS = Message 1 of the order 1887138700 at destination of +6512345678 has been sent at 09:28:45 (CET) on the 11-05-2005.	N
Delivered to Handset	Handset Ack	Message <MESSID> of the order <ORDERID> at destination of <MSISDNNBR> has been received at <TIME> (CET) on the <DATE>.	STATUS = Message 1 of the order 1887109085 at destination of +6512345678 has been received at 09:20:31 (CET) on the 11-05-2005.	N

Status, as displayed on the extranet	Notification Indicates	Message Sent to Customer	Customer Receives (Example)	Retriable Notification? (Y/N)
Temporary Negative				
DBF3: In retry	The platform is retrying delivery	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR>: Mobileway retrying.	STATUS = 0xDBF3 Message 1 of orderid 1886867226 for number +6512345678: Mobileway retrying.	N
Final Negative				
450D: Max amount of messages reached	Account limit is reached	<ERROR> You have reached the maximum amount of messages allowed. Please contact your account manager.	0x450D You have reached the maximum amount of messages allowed. Please contact your account manager.	N
MSISDN or destination operator blacklisted	MSISDN is blacklisted	Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> has been filtered on customer request.	STATUS = Message 1 of orderid 1886867226 for number +6512345678 has been filtered on customer request.	N
4503: No operator found for current MSISDN	No operator found for current MSISDN	<ERROR> Message <MESSID> of the order <ORDERID> at destination of <MSISDNNBR> is out of coverage	STATUS = 0x4503 Message 1 of the order 2016569242 at destination of +83118 is out of coverage	N
DB05: Call barred by operator	Call barred by operator	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> rejected by destination operator.	STATUS = 0xDB05 Message 1 of orderid 1886867226 for number +6512345678 rejected by destination operator.	N

Status, as displayed on the extranet	Notification Indicates	Message Sent to Customer	Customer Receives (Example)	Retriable Notification? (Y/N)
DB52: Failed message delivery	Failed Message Delivery	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> rejected by Operator.	STATUS = 0xDB52 Message 1 of orderid 1886867226 for number +6512345678 rejected by Operator.	N
DB76: Subscriber profile does not permit service	Subscriber profile does not permit service	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> rejected by Operator. MSISDN account is barred.	STATUS = 0xDB76 Message 1 of orderid 1886867226 for number +6512345678 rejected by Operator. MSISDN account is barred.	N
DB99: Unknown or Ported Number	Unknown or Ported Number	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> rejected by Operator.	STATUS = 0xDB99 Message 1 of orderid 1886867226 for number +6512345678 rejected by Operator.	N
DB62: Failed message delivery	Failed Message Delivery	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> rejected by Operator.	STATUS = 0xDB62 Message 1 of orderid 1886867226 for number +6512345678 rejected by Operator.	Y
DB65: Validity Period expired	Validity Period Expired	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> rejected by Operator.	STATUS = 0xDB65 Message 1 of orderid 1886867226 for number +6512345678 rejected by Operator.	Y
DB78: Subscriber is temporarily out of credit	Subscriber is temporarily out of credit	<ERROR> Message <MESSID> of orderid <ORDERID> for number <MSISDNNBR> does not have sufficient credit.	STATUS = 0xDB78 Message 1 of orderid 1886867226 for number +6512345678 does not have sufficient credit.	Y

4.2 Delivery Notifications in Detail

As described above, there are three levels of acknowledgments available in the HTTP interface:

- *SAP Ack*
- *SMS-C Ack*
- *Handset Ack*

The notification level is defined in the MT [SETUP] section.

4.2.1 SAP ACK

The SAP *Acceptance ACK* is generated when an MT message request is processed by the messaging gateway. Enterprises will always receive this acknowledgment, which includes the number of short messages created and the ORDERID for all.

4.2.2 SMS-C ACK

The *SMS-C ACK* is generated when an MT message has been sent to the destination mobile operator SMS-C and is waiting to be delivered to the handset. The SMS-C Ack therefore represents a temporary status (awaiting Handset Ack); although in cases where the Handset Ack is not requested, or not supported by the destination operator, the SMS-C Ack constitutes the final notification.

4.2.3 Handset Ack

The *Handset Ack* is generated when an MT message has been delivered to the mobile phone. This is an operator-dependent feature.

Table 12:summarizes the various acknowledgments obtained according to the two AckType options submitted by Enterprises in the MT message.

Table 12: Acknowledgements Received

AckType	Mobile Notification	Result
Order	No	Enterprise receives only SAP Ack
	Yes	Enterprise receives only SAP Ack but MobileNotification status will be tracked in database and available on the extranet.
Message	No	Enterprise receives notification of SAP and SMS-C Acks
	Yes	Enterprise receives notification of SAP, SMS-C, and Handset (where available) Acks

4.2.4 What is a Temporary Positive Notification Status?

A *Temporary Positive Notification* indicates to a customer an interim delivery confirmation, with a final delivery confirmation still outstanding.

4.2.5 What is a Final Positive Notification Status?

When the message has been delivered, a *Final Positive Notification* is returned A) to the operator SMS-C (if no Handset Ack is requested, or the operator does not support Handset Ack) or B) to the handset (if a Handset Ack is requested). In either case, the status is final and no additional notifications are sent for that message.

4.2.6 What is a Temporary Negative Notification Status?

A *Temporary Negative Notification* is returned when the platform is attempting a retry on a message that previously failed to deliver. When this happens, customers should expect a final status to be sent at the conclusion of the retry process, whether positive or negative. Until a final status is received, customers should not be initiating their own retry, as it could potentially result in duplicate messages arriving at the handset.

4.2.7 What is a Final Negative Notification Status?

A *Final Negative Notification* is returned when the platform could not deliver a particular message and will not attempt a retry. The returned status code indicates to the customer whether the message is *Retry-able* or not, in which case the customer can initiate their own retry process. There are no additional notifications sent after a Final Negative status.

4.2.8 What is a Retriable Status?

A *Retriable* status means that the platform will allow and or advise the customer to retry delivery of the message, although the plaform itself will not be attempting the retry.

4.2.8.1 Code DB62

Temporary Operator Network error ➔ Suggested retry scheme is to retry after 1min, 1 hour and 1 day after reception of this status code.

4.2.8.2 Code DB65

The operator SMSC was not able to deliver the message within its Validity Period.

4.2.8.3 Code DB78

End-user has no credit. Resend the message once end-user has topped up their account. You may want to send a free to end-user message, as a reminder.

4.2.9 What is a non Retriable Status?

A non *retriable* status means the platform disallow or advise the customer to not retry delivery of the message.

4.2.9.1 Code 4524

Meanings and actions:

- The end-user has requested not to be sent any SMS ➔ remove MSISDN from database.
- In the case of Premium SMS, the gateway blocks messages sent to an operator not configured on the account ➔ Check that the phone number is indeed correct and valid for the account used to deliver the message.
- The destination operator is blacklisted or not supported.

4.2.9.2 Code 4503

Meaning and action: The platform hasn’t found any operator associated to this MSISDN ➔ Check that MSISDN is correct.

4.2.9.3 Code DB05

The operator has blocked the message. This is not a network error but an explicit answer from the operator SMSC platform. For example it can be received when the end-user has reached is monthly/daily usage limit.

4.2.9.4 Code DB76

Upon reception of this error code customers must immediately stop the service provided to this MSISDN. DB76 can be due to user explicitly requesting to be barred from Premium services (i.e. corporate mobile phone contracts), or from using adult content, or recycled numbers. Non respect of subscriber's removal can result in punitive fines, short code cancellation or even legal action.

4.2.9.5 Code DB52

This is a final status code meaning there was a network error and the platform received a final error code. No retry is recommended as the same error is most likely to occur.

4.2.9.6 Code DB99

Unknown MSISDN or ported number.

5 DELIVERY NOTIFICATIONS FORMAT

Inside the acknowledgement "GET" request, data are separated by an ampersand (&). Each element is composed of a parameter and its value.

Table 13: Example Delivery Notification Request

Syntax
http://www.example.com.com/input_data.php3?CUSTOMERID=100&ORDERID=50&STATUS=STATUS_ENCODED &NBMESSAGE=1&SUBJECT=SUBJECT_ENCODED

There are ten different parameters available (Table 14:).

Table 14: Delivery Notification Request Available Parameters

Parameter	Value
CUSTOMERID	Identifier of the customer account.
ORDERID	Identifier of the request message sent by customer. For example, five MSISDNs are provided in the MT request, only one ORDERID is generated for this request.
STATUS	Status of notification data (ok/error).
MESSAGEID	Sequence Number of each message created when processing a MT request. For example, five MSISDNs are provided in the MT request, MESSAGEID for the first MSISDN is 1, for the second MSISDN is 2, all the way to 5 accordingly.
SUBJECT	Returns the field given in the MT request.
NBMESSAGE	Number of messages generated from the request.
MOBILEACK	Identifier of mobile notification. Only present when it is mobile notification. MOBILEACK=1
MSISDN	Destination phone number in international format (i.e., with “+”).
DATE	Notification date.
TIME	Notification time.
RETRY	Number of retries. If there are no retries, this will not be present.

5.1 Sample Delivery Notifications

SAP ACK Example
GET /MTAckReceive.aspx?CUSTOMERID=5061&ORDERID=1503644449&STATUS=Your%20message%20has%20been%20correctly%20processed.%201%20message%28s%29%20has%28have%29%20been%20generated%20with%20the%20order%20Id%201503644449.&NBMESSAGE=1&SUBJECT=Test_117036950_01333621357 HTTP/1.1 Host: 200.110.140.30
Decoded Status: STATUS= Your message has been correctly processed. 1 message(s) has (have) been generated with the order Id 1503644449.
SMS-C ACK Example
GET /MWC/jsp/s365Listener.jsp?CUSTOMERID=13937&ORDERID=1502431058&STATUS=Message%201%20of%20the%20order%201502431058%20at%20destination%20of%20%2B6512345678%20has%20been%20sent%20at%2005%3A32%3A56%20%28CET%29%20on%20the%2011-10-2004.&NBMESSAGE=1&SUBJECT=TSS&MESSAGEID=1&MSISDN=%2B6512345678&DATE=11-10-2004&TIME=05%3A32%3A56 HTTP/1.1
Decoded Status: STATUS= Message 1 of the order 1502431058 at destination of +6512345678 has been sent at 05:32:56 (CET) on the 11-10-2004.

SMS-C NAK (Failure) Example
GET /mw_notif?CUSTOMERID=13886&ORDERID=1504505254&STATUS=0xDB52%20Message%201%20of%20orderid%201504505254%20for%20number%20%2B447123456789%20rejected%20by%20Operator&NBMESSAGE=1&SUBJECT=tompax&MESSAGEID=1&MSISDN=%2B447123456789&DATE=15-10-2004&TIME=00%3A08%3A40 HTTP/1.1
Decoded Status: STATUS= 0xDB52 Message 1 of orderid 1504505254 for number +447123456789 rejected by Operator

Mobile Acknowledgement (Success) Example
GET /mms/notifManager.pl?CUSTOMERID=13502&ORDERID=1509312801&STATUS=Message%201%20of%20the%20order%201509312801%20at%20destination%20of%20%2B447967010269%20has%20been%20received%20at%2015%3A31%3A15%20%28CET%29%20on%20the%2015-10-2004.&NBMESSAGE=1&SUBJECT=447967010269-6FD1696299--SMS&MESSAGEID=1&MSISDN=%2B447967010269&DATE=15-10-2004&TIME=15%3A31%3A15&MOBILEACK=1 HTTP/1.1
Decoded Status: STATUS= Message 1 of the order 1509312801 at destination of +447967010269 has been received at 15:31:15 (CET) on the 15-10-2004.

5.2 Requesting optional parameters

The service provides three optional parameters which returns additional information on the status of a message. The parameters are available on demand.

Table 15: Delivery Notification Request Available Optional Parameters

Parameter	Key	Value
FINAL	A	Final Status Indicator: provides information on whether the delivery notification received is the final one or if the client should expect to receive further delivery notification(s). This is especially useful to know if a SMSC-Ack is the final ack or if a mobile ack is expected.
OPERATORID	B	Identifier of the destination operator as identified by SMS 365, enterprise service. Please contact your account representative should you need the list of operatorIDs.
INTERNALSTATUSID	C	Add additional information to the STATUSID provided in the delivery notification.
COUNTRYCODE	D	Country Code in ISO3166-2 format.
RECEIVED	E	Request Received Time (When request is received by the platform).
DELIVRED	F	Feedback request time (When delivery receipt is sent to Client) – the current time when DR is send to Client.
TPOA	G	TPOA / Shortcode / Longcode that was used for sending the message. This will be the final TPOA that was sent to the destination MNO.
GMT TIMESTAMP	H	Convert all the timestamps in DR to GMT.
MCC/MNC	M	Mobile Country Codes (MCC) and Mobile Network Codes (MNC).
FINAL DELIVERY NOTIFICATION	P	Only send the final delivery notification. Can be used in conjunction with parameter A.

5.2.1 Request format

To request optional parameters, clients need to add the string '?SAPF=XXXXXXXX' at the end of the AckReplyAddress of the MT request, where XXXXXXXX is either A, B, C, D, E, F, G, H or ABCDEFGH depending on what additional information clients wish to receive in the delivery notifications. The key (A, B, C, D, E, F, G, H) must be in upper case. The string '?SAPF=XXXXXXXX' will be stripped by the platform when submitting the GET requests for delivery notification. For instance if client has submitted 'http://example.com/api_php/MBW_DIR/notif.php?SAPF=ABC' in the MT request, the platform will send the GET requests to 'http://example.com/api_php/MBW_DIR/notif.php'.

5.2.1.1 Request for operatorID, final status indicator and internal status

```
[MSISDN]
List=+33619990000
```

```
[MESSAGE]
Text=Testing SMS from SAP. Have a nice day ahead!

[SETUP]
MobileNotification=Yes
AckType=MESSAGE
AckReplyAddress=http://example.com/api_php/MBW_DIR/notif.php?SAPF=ABC

[END]
```

5.2.1.2 Request for Operatorid only

```
[MSISDN]
List=+447123456789

[MESSAGE]
Text=Testing SMS from SAP. Have a nice day ahead!

[SETUP]
MobileNotification=Yes
AckType=MESSAGE
AckReplyAddress=http://example.com/api_php/MBW_DIR/notif.php?SAPF=B

[END]
```

5.2.1.3 Request for Internal Status only

```
[MSISDN]
List=+447123456789

[MESSAGE]
Text=Testing SMS from SAP. Have a nice day ahead!

[SETUP]
MobileNotification=Yes
AckType=MESSAGE
AckReplyAddress=http://example.com/api_php/MBW_DIR/notif.php?SAPF=C

[END]
```

5.2.1.4 Request for Country Code and TPOA only

```
[MSISDN]
List=+447123456789

[MESSAGE]
Text=Testing SMS from SAP. Have a nice day ahead!

[SETUP]
MobileNotification=Yes
AckType=MESSAGE
AckReplyAddress=http://example.com/api_php/MBW_DIR/notif.php?SAPF=DG

[END]
```

5.2.2 Understanding the responses

SMS-C NAK (Failure, Final) Example with SAPF=AB
GET /notif.php?CUSTOMERID=13886&ORDERID=1504505254&STATUS=0xDB52%20Message%201%20of%20orderid%201504505254%20for%20number%20%2B447123456789%20rejected%20by%20Operator&NBMESSAGE=1&SUBJECT=tompax&MESSAGEID=1&MSISDN=%2B447123456789&DATE=15-10-2004&TIME=00%3A08%3A40&FINAL=Y&OPERATORID=25 HTTP/1.1
Optional parameter information explained: The message has been rejected by the SMSC, no further Delivery Notification will be sent. The destination operatorID is 25 (Vodafone Ireland(EIRCELL))

SMS-C ACK Example (positive, final) example with SAPF=A
GET /MWC/jsp/s365Listener.jsp?CUSTOMERID=13937&ORDERID=1502431058&STATUS=Message%201%20of%20the%20order%201502431058%20at%20destination%20of%20%2B6512345678%20has%20been%20sent%20at%2005%3A32%3A56%20%28CET%29%20on%20the%2011-10-2004.&NBMESSAGE=1&SUBJECT=TSS&MESSAGEID=1&MSISDN=%2B6512345678&DATE=11-10-2004&TIME=05%3A32%3A56&FINAL=Y HTTP/1.1
Optional parameter information explained: The message has been delivered to the SMSC. No further delivery notifications wil be submitted.

SMS-C ACK Example (positive, intermediate) example with SAPF=AB

```
GET /MWC/jsp/s365Listener.jsp?CUSTOMERID=13937&ORDERID=1502431
058&STATUS=Message%201%20of%20the%20order%201502431058%20at%20destination%20of%20%2B6512345
678%20has%20been%20sent%20at%2005%3A32%3A56%20%28CET%29%20on%20the%2011-10-2004.&NB
MESSAGE=1&SUBJECT=TSS&MESSAGEID=1&MSISDN=%2B6512345678&DATE=11-10-
2004&TIME=05%3A32%3A56&FINAL=N& OPERATORID=7
 HTTP/1.1
```

Optional parameter information explained:

The message has been delivered to the SMSC, waiting for mobile acknowledgement.

Mobile Acknowledgement (Success, final) example with SAPF=AB

```
GET
/mms/notifManager.pl?CUSTOMERID=13502&ORDERID=1509312801&STATUS=Message%201%20of%20the%20o
rder%201509312801%20at%20destination%20of%20%2B447967010269%20has%20been%20received%20at%2015
%3A31%3A15%20%28CET%29%20on%20the%2015-10-2004.&NBMESSAGE=1&SUBJECT=447967010269-
6FD1696299--SMS&MESSAGEID=1&MSISDN=%2B447967010269&DATE=15-10-
2004&TIME=15%3A31%3A15&MOBILEACK=1 &FINAL=Y&OPERATORID=34
 HTTP/1.1
```

Optional parameter information explained:

Message was delivered to the handset. No further status will be provided.

The destination operatorID is 34 (France Telecom Espana(ORANGE SPAIN))

SMS-C NAK (Failure, Final) Example with SAPF=ABC

```
GET
/notif.php?CUSTOMERID=13886&ORDERID=1504505254&STATUS=0xDB52%20Message%201%20of%20orderid
%201504505254%20for%20number%20%2B447123456789%20rejected%20by%20Operator&NBMESSAGE=1&SUB
JECT=tompax&MESSAGEID=1&MSISDN=%2B447123456789&DATE=15-10-
2004&TIME=00%3A08%3A40&FINAL=Y&OPERATORID=25 &INTERNALSTATUS=DB04
 HTTP/1.1
```

Optional parameter information explained:

The message has been rejected by the SMSC, there was an operator connection error.

No further Delivery Notification will be sent.

The destination operatorID is 25 (Vodafone Ireland (EIRCELL)).

SMS-C ACK Example (positive, final) example with SAPF=AC

```
GET /MWC/jsp/s365Listener.jsp?CUSTOMERID=13937&ORDERID=1502431
058&STATUS=Message%201%20of%20the%20order%201502431058%20at%20destination%20of%20%2B6512345
678%20has%20been%20sent%20at%2005%3A32%3A56%20%28CET%29%20on%20the%2011-10-2004.&NB
MESSAGE=1&SUBJECT=TSS&MESSAGEID=1&MSISDN=%2B6512345678&DATE=11-10-
2004&TIME=05%3A32%3A56&FINAL=Y &INTERNALSTATUS=450A
 HTTP/1.1
```

Optional parameter information explained:

The message has been delivered to the SMSC. No further delivery notifications wil be submitted.

Mobile Acknowledgement (Success, final) example with SAPF=ABCDEFGH

```
GET
/mms/notifManager.pl?CUSTOMERID=2655&ORDERID=1671583028&STATUS=Message%2012%20of%20the%20o
rder%201671583028%20at%20destination%20of%20%2B85291033386%20has%20been%20received%20at%2016
%3A32%3A24%20(GMT)%20on%20the%2016-04-2013.&NBMESSAGE=20&SUBJECT= 447967010269-
6FD1696299--SMS&MESSAGEID=12&MSISDN=%2B85291033386&DATE=16-04-
2016&TIME=11%3A32%3A24&MOBILEACK=1&FINAL=Y&OPERATORID=45&INTERNALSTATUS=1234567890&C
OUNTRYCODE=HK&TPOA=CitibankHK%20IOOO&DELIVRED=20160416115310&RECEIVED=20160416113224
HTTP/1.1
```

Optional parameter information explained:

Message was delivered to the handset. No further status will be provided.

The destination operatorID is 34 (France Telecom Espana(ORANGE SPAIN))

Country Code returned is HK (ISO 3166-2)

TPOA returned CitibankHK

DELIVRED will return the time when the DR is sent

RECEIVED will return the time when the request is received by SAP

All timestamp are in GMT (as SAPF=H)

5.2.3 INTERNALSTATUSID codes list

The list of all available INTERNALSTATUSID codes, how they are registered in our system and their corresponding STATUSID is displayed in the table below.

Internal StatusID	MT Status Description	External StatusID
4503	No SMSC found for current MSISDN	4503
4507	Msg received by SMSC	450A
4510	Msg processed by the platform.	N/A
4518	No SMSC found to send UDH	DB52
4524	Trashed Messages	4524
4525	Trashed Messages on customer request	4524
4526	Unknown operator	DB52
4527	Unknown SMSC	DB52
4529	NRS error	DB62
4530	Customer Blacklisted	4524
4533	Message been blacklisted due to bad sender ID	4533
4534	Message has been trashed on customer request	4534

Internal StatusID	MT Status Description	External StatusID
450A	Msg received	450A
450D	Msg refused by billing	450D
450E	Msg received by the plaform	450E
451A	Handset Ack	451A
DA10	Invalid MSISDN	DB52
DA11	Invalid TPOA	DB52
DB02	SYNTAX_ERROR	DB52
DB03	SMS service not available	DB52
DB04	Operator connection error	DB52
DB05	Call barred by destination operator	DB05
DB07	AUTH_FAILURE	DB52
DB11	REPET_LEG_FAILURE	DB52
DB16	REVERS_CHARG_NOT_ALLOWED	DB78
DB19	NEW_AC_NOT_VALID	DB52
DB22	TIME_PERIOD_NOT_VALID	DB52
DB23	MSG_TYPE_NOT_SUPPORTED	DB52
DB24	MSG_TOO_LONG	DB52
DB35	Customer busy	DB52
DB36	RPID_ALREADY_IN_USE	DB52
DB37	DELIVERY_IN_PROGRESS	N/A
DB42	NO_MSG_ASSOCIATED	DB52
DB43	TELESERVICE_NOT_PROVISIONNED	DB52
DB44	MEMORY_CAPACITY_EXCEEDED	DB62
DB46	VALIDITY_PERIOD_NOT_VALID	DB52
DB47	ADC_INVALID	DB99
DB48	Invalid MSISDN	DB99
DB51	INVALID_PWD_ID	DB52
DB52	MSG_SEND_FAILURE	DB52
DB53	SERVER_BUSY	DB62
DB54	TOO_MANY_CONNECTION	DB62
DB55	Operator network error	DB52
DB56	STATUS_UNKNOW	DB52
DB58	INVALID_SIZE	DB52

Internal StatusID	MT Status Description	External StatusID
DB60	Data format not supported	DB52
DB62	OTHER_FAILURE_REASON	DB62
DB63	Unknown subscriber	DB99
DB64	FACILITY_NOT_SUPPORTED	DB52
DB65	Validity period expired	DB65
DB66	SMS_NOT_PROVISIONED	DB52
DB67	MS_ERROR	DB52
DB69	Brite subscriber	DB76
DB70	Distant operator network failure	DB52
DB71	NO_RESPONSE_FROM_SMSC	DB52
DB73	CONNECTION_TO_SMSC_FAILED	DB52
DB74	BAD_REQUEST	DB52
DB76	INVALID_SUBSCRIPTION	DB76
DB77	SERVER_INTERNAL_ERROR	DB62
DB78	BILLING_NO_BALANCE	DB78
DB80	ACK_ONLY	DB62
DB81	BLACKLISTED	DB52
DB85	Message Validity Period Expired (KGT)	DB65
DB99	unknown or ported number	DB99
DBA3	Handset cannot be reached (Sunday)	DB52
DBA4	Subscriber cannot be reached (Sunday)	DB52
DBA5	Subscriber SIM Card Full (Sunday)	DB62
DBA6	SMSC currently busy	DB62
DBA7	Temporary Traffic Congestion (Sunday)	DB62
DBF1	Message rejected by the platform Output Interface. Message not sent to Operator	DB52
DBF2	Message rejected by Operator. This a generic error	DB52
DBF3	Retry currently done on message	DBF3
DBF5	End User roamed on an unreachable network	DB52
DBF6	Message rejected by Operator. User must send MO to unblock messages	DBF6
DBF7	Message rejected by Operator. Exceed more than 50MT per month not allowed	DB52
DBF8	Message rejected by Operator. End user not registered for the service	DB76
DC16	DLT Entity Error	DC16

Internal StatusID	MT Status Description	External StatusID
DC17	DLT Template Error	DC17
DC18	DLT General Error	DC18

5.3 Delivery Notifications - EnterpriseResponse

The Enterprise notification URL must respond to the request with a positive HTTP 200 OK response. Otherwise, the messaging gateway continues to send the message, assuming that the initial attempt was unsuccessful.

Message Acknowledgment Example
HTTP/1.1 200 OK Content-Length: 2 Content-Type: text/html OK
HTTP/1.1 200 OK Content-Length: 3 Content-Type: text/html NOK

If the messaging gateway doesn't receive a 200 response, it will retry up to 9 times, every 30 minutes.

Important: Due to the various retries, Enterprises must implement a system that will support receiving notifications in an incorrect order. i.e. Handset Ack might be received before SMSC Ack.

6 MO HTTP INTERFACE

The HTTP messaging gateway expects Enterprises to implement an HTTP server or Web service to receive MO message via HTTP interface.

6.1 HTTP Request for MO Messages

The HTTP messaging gateway initiates an HTTP MO request to an Enterprises‘ URL when there is an MO message waiting to be delivered to a certain Customer Account. Please note that the Enterprise’s firewall must be open to the HTTP messaging gateway IP before receiving any MO messages.

The Message Body is in SCS encoding for all characters but the ones listed in Appendix A, Part 0.

An MO message is a URL-encoded XML request. The example below contains a basic MO request:

MO Request Example

```
POST /customer_file_access HTTP/1.1
HOST: customers_server
Authorization: Basic xxxxxxxx (optional)
Content-Length: 917

XmlMsg=%3C%3Fxml%20version%3D%221.0%22%20%3F%3E%0A%3CSMS_MO%3E%0A%3CMSISDN%3E%2B
1234567890%3C%2FMSISDN%3E%0A%3CORIGINATING_ADDRESS%3E46645%3C%2FORIGINATING_ADDRE
SS%3E%0A%3CMESSAGE%3EPep%20boys.%2019464%3C%2FMESSAGE%3E%0A%3CPARAMETERS%3E%0A
%3COPERATORID%3E787%3C%2FOPERATORID%3E%0A%3CACCOUNTID%3E13909%3C%2FACCOUNTID%3
E%0A%3CMESSAGEID%3E242922646%3C%2FMESSAGEID%3E%0A%3COPERATOR_INFORMATION%3E%0A
%3COPERATOR_STANDARD%3EGSM%3C%2FOPERATOR_STANDARD%3E%0A%3COPERATOR_CODE%3E
%0A%3CMCC%3EN%2FA%3C%2FMCC%3E%0A%3CMNC%3EN%2FA%3C%2FMNC%3E%0A%3C%2FOPERAT
OR_CODE%3E%0A%3C%2FOPERATOR_INFORMATION%3E%0A%3CDCS%3E7b%3C%2FDCS%3E%0A%3CCL
ASS%3E2%3C%2FCLASS%3E%0A%3CRECEIVED_SERVICENUMBER%3E46645%3C%2FRECEIVED_SERVICE
NUMBER%3E%0A%3CKEYWORD%3EN%2FA%3C%2FKEYWORD%3E%0A%3CRECEIVEDTIME%3E%0A%3CD
ATE%3EThu%2C%2014%20Oct%202004%3C%2FDATE%3E%0A%3CTIME%3E00%3A17%3A43%3C%2FTIME%3
E%0A%3C%2FRECEIVEDTIME%3E%0A%3C%2FPARAMETERS%3E%0A%3C%2FSMS_MO%3E
```

MO Request Example, Continued

XmlMsg is URL-encoded, after decoding:

```
<?xml version="1.0" ?>
<SMS_MO>
   <MSISDN>+1234567890</MSISDN>
   <ORIGINATING_ADDRESS>46645</ORIGINATING_ADDRESS>
   <MESSAGE>Pep boys. 19464</MESSAGE>
   <PARAMETERS>
      <OPERATORID>787</OPERATORID>
      <ACCOUNTID>13909</ACCOUNTID>
      <MESSAGEID>242922646</MESSAGEID>
      <OPERATOR_INFORMATION>
      <OPERATOR_STANDARD>GSM</OPERATOR_STANDARD>
         <OPERATOR_CODE>
            <MCC>N/A</MCC>
            <MNC>N/A</MNC>
         </OPERATOR_CODE>
      </OPERATOR_INFORMATION>
      <DCS>7b</DCS>
      <CLASS>2</CLASS>
      <RECEIVED_SERVICENUMBER>46645</RECEIVED_SERVICENUMBER>
       <KEYWORD>N/A</KEYWORD>
      <RECEIVEDTIME>
         <DATE>Thu, 14 Oct 2004</DATE>
         <TIME>00:17:43</TIME>
      </RECEIVEDTIME>
   </PARAMETERS>
</SMS_MO>
```

The following is an example MO with the SESSION_ID and TAC Code:

MO with SESSION_ID and TAC Code Example

```
POST /customer_file_access HTTP/1.1
HOST: customers_server
Authorization: Basic xxxxxxxx (optional)
Content-Length: 941

XmlMsg=%3C?xml%20version=%221.0%22%20?%3E%0A%3CSMS_MO%3E%0A%3CMSISDN%3E+60123456789
%3C/MSISDN%3E%0A%3CORIGINATING_ADDRESS%3E36999%3C/ORIGINATING_ADDRESS%3E%0A%3CME
SSAGE%3EGENY%2024630757%3C/MESSAGE%3E%0A%3CPARAMETERS%3E%0A%3COPERATORID%3E228
%3C/OPERATORID%3E%0A%3CACCOUNTID%3E9%3C/ACCOUNTID%3E%0A%3CMESSAGEID%3E198061648
%3C/MESSAGEID%3E%0A%3COPERATOR_INFORMATION%3E%0A%3COPERATOR_STANDARD%3EGSM%3
C/OPERATOR_STANDARD%3E%0A%3COPERATOR_CODE%3E%0A%3CMCC%3E502%3C/MCC%3E%0A%3C
MNC%3E19%3C/MNC%3E%0A%3C/OPERATOR_CODE%3E%0A%3C/OPERATOR_INFORMATION%3E%0A%3C
DCS%3E7b%3C/DCS%3E%0A%3CCLASS%3E2%3C/CLASS%3E%0A%3CTACCODE%3Etaccodevalue%3C/TACC
ODE%3E%0A%3CSESSION_ID%3ESESSION_IDvalue%3C/SESSION_ID%3E%0A%3CRECEIVED_SERVICENUM
BER%3E36999%3C/RECEIVED_SERVICENUMBER%3E%0A%3CKEYWORD%3EGENY%3C/KEYWORD%3E%0A
%3CRECEIVEDTIME%3E%0A%3CDATE%3EFri,%209%20Apr%202004%3C/DATE%3E%0A%3CTIME%3E15:18:2
2%3C/TIME%3E%0A%3C/RECEIVEDTIME%3E%0A%3C/PARAMETERS%3E%0A%3C/SMS_MO%3E
```

MO with SESSION_ID and TAC Code Example, Continued

XmlMsg is URL-encoded, after decoding:

```
<?xml version="1.0" ?>
<SMS_MO>
<MSISDN>+60123456789</MSISDN>
<ORIGINATING_ADDRESS>36999</ORIGINATING_ADDRESS>
<MESSAGE>GENY 24630757</MESSAGE>
<PARAMETERS>
<OPERATORID>228</OPERATORID>
<ACCOUNTID>9</ACCOUNTID>
<MESSAGEID>198061648</MESSAGEID>
<OPERATOR_INFORMATION>
<OPERATOR_STANDARD>GSM</OPERATOR_STANDARD>
<OPERATOR_CODE>
<MCC>502</MCC>
<MNC>19</MNC>
</OPERATOR_CODE>
</OPERATOR_IN
FORMATION>
<DCS>7b</DCS>
<CLASS>2</CLASS>
<TACCODE>taccodevalue</TACCODE>
<SESSION_ID>SESSION_IDvalue</SESSION_ID>
<RECEIVED_SERVICENUMBER>36999</RECEIVED_SERVICENUMBER>
<KEYWORD>GENY</KEYWORD>
<RECEIVEDTIME>
<DATE>Fri, 9 Apr 2004</DATE>
<TIME>15:18:22</TIME>
</RECEIVEDTIME>
</PARAMETERS>
</SMS_MO>
```

6.1.1 MO Message XML DTD

The HTTP messaging Gateway can POST an MO message to Enterprises using the following MO Message XML DTD:

```
<! -- DTD of the sms_mo.xml file -->

<! ENTITY % URI "CDATA"> <! -- URI designating a hypertext node -->

<!ELEMENT SMS_MO (MSISDN, ORIGINATING_ADDRESS, MESSAGE, PARAMETERS)>
<!ELEMENT MSIS DN (#PCDATA)>
<!ELEMENT ORIGINATING_ADDRESS (#PCDATA)>
<!ELEMENT MESSAGE (#PCDATA)>
<!ELEMENT PARAMETERS (OPERATORID, ACCOUNTID, MESSAGEID, OPERATOR_INFORMATION, DCS,
CLASS, TACCODE?, SESSION_ID?, UDH?, RECEIVED_SERVICENUMBER, KEYWORD , RECEIVEDTIME)>
<!ELEMENT OPERATORID (#PCDATA)>
<!ELEMENT ACCOUNTID (#PCDATA)>
<!ELEMENT MESSAGEID (#PCDATA)>
<!ELEMENT OPERATOR_INFORMATION (OPERATOR_STANDARD, OPERATOR_CODE)>
<!ELEMENT OPERATOR_STANDARD (#PCDATA)>
<!ELEMENT OPERATOR_CODE (MCC, MNC)>
<!ELEMENT MCC (#PCDATA)>
<!ELEMENT MNC (#PCDATA)>
<!ELEMENT DCS (#PCDATA)>
<!ELEMENT CLASS (#PCDATA)>
<!ELEMENT TACCODE (#PCDATA)>
<!ELEMENT SESSION_ID (#PCDATA)>
<!ELEMENT UDH (#PCDATA)>
<!ELEMENT RECEIVED_SERVICENUMBER (#PCDATA)>
<!ELEMENT KEYWORD (#PCDATA)>
<!ELEMENT RECEIVEDTIME (DATE, TIME)>
<!ELEMENT DATE (#PCDATA)>
<!ELEMENT TIME (#PCDA TA)>
```

6.1.2 Message XML Tag Definition

Table 16: below explains each XML tag used in the MO request.

Table 16: MO Message XML Tag Definitions

MO XML Elements	
Element	**Value**
SMS_MO	The **SMS_MO** is a root element, and contains following child elements: **MSISDN**, **ORIGINATING_ADDRESS**, **MESSAGE**, **PARAMETERS**
MSISDN	The **MSISDN** field contains the phone number of the SMS originator. The phone number is in international format, i.e. with '+' at the beginning, followed by country code and phone number.
ORIGINATING_ADDRESS	The **ORIGINATING_ADDRESS** field contains the short code or service number dedicated to an Enterprise account. An Enterprise can use this value as the OriginatingAddress, under [SETUP] section in the MT request.
MESSAGE	The **MESSAGE** field contains the text sent by the mobile subscriber. The message could be in SAP encoding (refer to appendix) if it is a text message or in hexadecimal string if it is an 8-bit (Binary) or UCS2 message. The specific markers <CR> (carriage return) and <LF> (line feed) are used to insert the characters respectively. These markers are inserted as **&lt;CR&gt;** or **&lt;LF&gt;** inside the XML element using HTML encoding (refer to Appendix A: SAP Character Set for all special characters)
PARAMETERS	The **PARAMETERS** field contains following child elements: **OPERATOR_INFORMATION**, **DCS**, **CLASS**, **UDH** (optional), **SESSION_ID** (Optional), **KEYWORD**, **RECEIVEDTIME**
OPERATORID	The **OPERATORID** field specifies the identifier of the mobile operator to which the MSISDN belongs. The values are the platform internal values and are given in decimal format. These are described in the Operator List document.
ACCOUNTID	The **ACCOUNTID** field specifies the platform unique identifier of the customer account. This value is used to differentiate accounts when a customer has different accounts with us. This parameter is given in decimal format.
MESSAGEID	The **MESSAGEID** field specifies the platform unique identifier of the MO message. It can be used to detect duplicates. If receiving the same messageID for another message the content provider platform should accept the message but trash it as it is a duplicate. This parameter is given in decimal format.
OPERATOR_INFORMATION	The **OPERATOR_INFORMATION** field contains the information about the subscriber's operator. It contains following child elements: **OPERATOR_STANDARD**, **OPERATOR_CODE**
OPERATOR_STANDARD	The **OPERATOR_STANDARD** field specifies the standard of the mobile operator. Possible values are: **GSM, GSM_1900**, **GSM_1800**, **GSM_3G**, **FIXED_LINE**, **CDMA**, **TDMA**, **IDEN**, **PCS**, **UMTS**, **AMPS**, **UNKNOWN**, **N/A**.

MO XML Elements	
Element	**Value**
OPERATOR_CODE	The **OPERATOR_CODE** field contains following child elements: **MNC** and **MCC**
MNC	The **MNC** field specifies the Mobile Network Code of the operator in decimal format. If no MNC is applicable to the operator, this field value is **N/A**.
MCC	The **MCC** field specifies the Mobile Country Code of the operator in decimal format. If no MCC is applicable to the operator, this field value is **N/A**.
DCS	The **DCS** field specifies how the SMS message is encoded. Possible values include: • **7b** (representing SAP default alphabet) • **8b** (representing 8-bit binary message) • **UCS2**
CLASS	The **CLASS** field specifies the received type of short message. There are four possible values: **0**, **1**, **2**, and **3**. The Class values are described in the MT section above.
TACCODE (Optional)	If displayed, the **TACCODE** field contains the TAC (Type Allocation Code), which is a part of the IMEI (International Mobile station Equipment Identity), from the sending operator. It enables the content provider to identify the handset's terminal type.
SESSION_ID (Optional)	The **SESSION_ID** field is used for session tracking or other purposes. When this field is present, its contents must be transferred directly to the SESSION_ID field for each MT associated with this MO. (Depending on the service, this field may be used for other information. If so, this will be discussed by the account manager.)
UDH (Optional)	The **UDH** field contains the UDH sent by the mobile subscriber when its message contains one. When there is no UDH inside the MO message, this tag is not present. The UDH is given in hexadecimal format.
RECEIVED_SERVICENUMBER	The **RECEIVED_SERVICENUMBER** field contains the service number sent by the mobile operator. Note: This parameter may be different from the Originating_Address. *Example:* For short code *1229*, Enterprise should receive Originating_Address as short code *1229*. The operator may send *12291* or *12292* depending on the operator's SMS-C configuration. In this case, the real value from the operator is stored inside the Received_ServiceNumber tag.
KEYWORD	The **KEYWORD** field contains the keyword used to parse the body of the message; this element is not inserted inside the request if the MO is not parsed.
RECEIVEDTIME	The **RECEIVEDTIME** field contains following child elements: **DATE**, **TIME**

MO XML Elements	
Element	**Value**
DATE	The **DATE** field specifies the date of receipt and is formatted as follows: Ddd, dd mmm yyyy Where: *Ddd=day of the week (Mon, Tue, Wed, …)* *dd=day* *mmm=month (Jan, Feb, Mar, …)* *yyyy=year* Note: Time is Central European Time (CET or CEST).
TIME	The **TIME** element qualifies the time at which the SMS is received and is formatted as follows: hh:mm:ss Where: *hh=hour* *mm=minute* *ss=second* Note: Time is Central European Time (CET or CEST).

6.2 HTTP Response for MO Messages

The Enterprise HTTP Server must respond to the MO request with a positive HTTP 200 OK response. Otherwise, the messaging gateway continues to send the message, assuming that the initial attempt was unsuccessful. In the case the messaging gateway doesn’t receive a positive response it will retry up to 9 times, every 30 minutes.

HTTP Responses for MO Messages Example
HTTP/1.1 200 OK Content-Length: 2 Content-Type: text/html OK
HTTP/1.1 200 OK Content-Length: 3 Content-Type: text/html NOK

6.3 MO Messages Retry Interval

In the case the messaging gateway doesn’t receive a positive response it will retry up to 9 times, every 30 minutes by default. There are other retry options available but this would require a request to your account manager to configure.

Retry Interval	
5 seconds	Message will retry every 5 seconds
10 seconds	Message will retry every 10 seconds
15 seconds	Message will retry every 15 seconds
20 seconds	Message will retry every 20 seconds
25 seconds	Message will retry every 25 seconds
30 seconds	Message will retry every 30 seconds
10 minutes	Message will retry every 10 minutes
15 minutes	Message will retry every 15 minutes
30 minutes (default)	Message will retry every 30 mintues

7 HTTP SERVER ADDITIONAL FEATURES

A special page has been created to test messages, accounts, and connectivity without sending an actual message. The messaging gateway validates all messages sent to this page and responds accordingly to help Enterprises troubleshoot their HTTP interface implementation.

The following is the testing page to access */customerfolder/Message-test.sms*.

HTTP Interface Troubleshooting Example

```
POST /customerfolder/Message-test.sms HTTP/1.1
HOST: sms-pp.sapmobileservices.com
Authorization: Basic TW9iaWxlMzY1Ok0zNjU=
Content-Length: 95

Subject=Example
[MSISDN]
List=+44777123123,+44777123124,+44777123125
[MESSAGE]
Text=A TESTING SMS to three mobile phones.
[END]
```

8 APPENDIX A: SAP CHARACTER SET

SMS 365, Enterprise Service Messaging Platform expects to receive the values listed in column “SAP Encoding”

Default Alphabet Character Table

Description	Visual Character on the phone	SAP Encoding
SPACE		20
EXCLAMATION MARK	!	21
QUOTATION MARK	"	22
NUMBER SIGN	#	23
DOLLAR SIGN	$	24
PERCENT SIGN	%	25
AMPERSAND	&	26
APOSTROPHE	'	27
LEFT PARENTHESIS	(	28
RIGHT PARENTHESIS	)	29
DIGIT ZERO	0	30
DIGIT ONE	1	31
DIGIT TWO	2	32
DIGIT THREE	3	33
DIGIT FOUR	4	34
DIGIT FIVE	5	35
DIGIT SIX	6	36
DIGIT SEVEN	7	37
DIGIT EIGHT	8	38
DIGIT NINE	9	39
COMMERCIAL AT	@	40
LATIN CAPITAL LETTER A	A	41
LATIN CAPITAL LETTER B	B	42
LATIN CAPITAL LETTER C	C	43
LATIN CAPITAL LETTER D	D	44
LATIN CAPITAL LETTER E	E	45
LATIN CAPITAL LETTER F	F	46
LATIN CAPITAL LETTER G	G	47
LATIN CAPITAL LETTER H	H	48
LATIN CAPITAL LETTER I	I	49

Description	Visual Character on the phone	SAP Encoding
LATIN CAPITAL LETTER P	P	50
LATIN CAPITAL LETTER Q	Q	51
LATIN CAPITAL LETTER R	R	52
LATIN CAPITAL LETTER S	S	53
LATIN CAPITAL LETTER T	T	54
LATIN CAPITAL LETTER U	U	55
LATIN CAPITAL LETTER V	V	56
LATIN CAPITAL LETTER W	W	57
LATIN CAPITAL LETTER X	X	58
LATIN CAPITAL LETTER Y	Y	59
LATIN SMALL LETTER A	a	61
LATIN SMALL LETTER B	b	62
LATIN SMALL LETTER C	c	63
LATIN SMALL LETTER D	d	64
LATIN SMALL LETTER E	e	65
LATIN SMALL LETTER F	f	66
LATIN SMALL LETTER G	g	67
LATIN SMALL LETTER H	h	68
LATIN SMALL LETTER I	i	69
LATIN SMALL LETTER P	p	70
LATIN SMALL LETTER Q	q	71
LATIN SMALL LETTER R	r	72
LATIN SMALL LETTER S	s	73
LATIN SMALL LETTER T	t	74
LATIN SMALL LETTER U	u	75
LATIN SMALL LETTER V	v	76
LATIN SMALL LETTER W	w	77
LATIN SMALL LETTER X	x	78
LATIN SMALL LETTER Y	y	79
LINE FEED	LF	3C4C463E
CARRIAGE RETURN	CR	3C43523E
ASTERISK	*	2A
PLUS SIGN	+	2B
COMMA	,	2C

Description	Visual Character on the phone	SAP Encoding
HYPHEN-MINUS	-	2D
FULL STOP	.	2E
FORWARD SLASH	/	2F
COLON	:	3A
SEMICOLON	;	3B
LESS THAN	<	3C
EQUALS	=	3D
GREATER THAN	>	3E
QUESTION MARK	?	3F
LATIN CAPITAL LETTER J	J	4A
LATIN CAPITAL LETTER K	K	4B
LATIN CAPITAL LETTER L	L	4C
LATIN CAPITAL LETTER M	M	4D
LATIN CAPITAL LETTER N	N	4E
LATIN CAPITAL LETTER O	O	4F
LATIN CAPITAL LETTER Z	Z	5A
LOW LINE	_	5F
LATIN SMALL LETTER J	j	6A
LATIN SMALL LETTER K	k	6B
LATIN SMALL LETTER L	l	6C
LATIN SMALL LETTER M	m	6D
LATIN SMALL LETTER N	n	6E
LATIN SMALL LETTER O	o	6F
LATIN SMALL LETTER Z	z	7A
INVERTED EXCLAMATION POINT	¡	A1
POUND STERLING	£	A3
GENERAL CURRENCY SYMBOL	¤	A4
YEN	¥	A5
SECTION SIGN	§	A7
INVERTED QUESTION MARK	¿	BF
LATIN CAPITAL LETTER A WITH DIAERESIS	Ä	C4
LATIN CAPITAL LETTER A WITH RING ABOVE	Å	C5
LATIN CAPITAL LETTER AE	Æ	C6
LATIN CAPITAL LETTER C WITH CEDILLA	Ç	C7

Description	Visual Character on the phone	SAP Encoding
LATIN CAPITAL LETTER E WITH ACUTE	É	C9
LATIN CAPITAL LETTER N WITH TILDE	Ñ	D1
LATIN CAPITAL LETTER O WITH DIAERESIS	Ö	D6
LATIN CAPITAL LETTER O WITH STROKE	Ø	D8
LATIN CAPITAL LETTER U WITH DIAERESIS	Ü	DC
LATIN SMALL LETTER SHARP S	ß	DF
LATIN SMALL LETTER A WITH GRAVE	à	E0
GREEK CAPITAL LETTER DELTA	Δ	E1
GREEK CAPITAL LETTER THETA	Θ	E2
LATIN SMALL LETTER A WITH DIAERESIS	ä	E4
LATIN SMALL LETTER A WITH RING ABOVE	å	E5
LATIN SMALL LETTER AE	æ	E6
LATIN SMALL LETTER E WITH GRAVE	è	E8
LATIN SMALL LETTER E WITH ACUTE	é	E9
GREEK CAPITAL LETTER LAMDA	Λ	EA
GREEK CAPITAL LETTER XI	Ξ	EB
LATIN SMALL LETTER I WITH GRAVE	ì	EC
GREEK CAPITAL LETTER PI	Π	ED
GREEK CAPITAL LETTER PHI	Φ	EE
GREEK CAPITAL LETTER PSI	Ψ	EF
LATIN SMALL LETTER N WITH TILDE	ñ	F1
LATIN SMALL LETTER O WITH GRAVE	ò	F2
GREEK CAPITAL LETTER GAMMA	Γ	F3
LATIN SMALL LETTER O WITH DIAERESIS	ö	F6
LATIN SMALL LETTER O WITH STROKE	ø	F8
LATIN SMALL LETTER U WITH GRAVE	ù	F9
GREEK CAPITAL LETTER OMEGA	Ω	FA
GREEK CAPITAL LETTER SIGMA	Σ	FB
LATIN SMALL LETTER U WITH DIAERESIS	ü	FC

GSM 7-bit Default Alphabet Extension Table
This code is an escape to an extension of the 7-bit default alphabet table. A receiving entity that does not understand the meaning of this escape mechanism should display it as a space character.

The characters in this table are countered as 2 bytes hence actual content length will differ.

Description	Visual Character on the phone	SAP hex Encoding
LEFT CURLY BRACKET	{	FF28 or 7B
RIGHT CURLY BRACKET	}	FF29 or 7D
BACK SLASH	\	FF2F
LEFT SQUARE BRACKET	[	FF3C or 5B
TILDE	~	FF3D
RIGHT SQUARE BRACKET	]	FF3E or 5D
EURO SIGN	€	FF65 or 80
VERTICAL LINE	|	FFA1 or 7C
CIRCUMFLEX ACCENT	^	FFEA or 5E
If this can’t be displayed, a ‘\n’ should be used.	Page Break Character	FFF9
This is the escape character for the extended GSM default 7 bit extension character set. This will be used as an extension to the extension library.	N/A	FF

MO interface

The Messaging Platform sends the first value in column "SAP hex encoding". E.g. character € will be encoded as FF65. Due to the XML nature of the interface some characters have to be HTML encoded as below:

Description	Visual Character on the phone	SAP hex Encoding	Latin-1 display
AMPERSAND	&	26616D703B	&amp;
QUOTATION MARK	"	2671756F74743B	&quot;
APOSTROPHE	'	2661706F733B	&apos;
SPACE		266E62736F3B	&nbsp;
LESS THAN	<	266C743B	&lt;
GREATER THAN	>	2667743B	&gt;

9 APPENDIX B: GREEK CHARACTERS IN LATIN/GREEK 8859-7

Character	Hex	Entity
Greek capital letter alpha	C1	Α
Greek capital letter beta	C2	Β
Greek capital letter gamma	C3	Γ
Greek capital letter delta	C4	Δ
Greek capital letter epsilon	C5	Ε
Greek capital letter zeta	C6	Ζ
Greek capital letter eta	C7	Η
Greek capital letter theta	C8	Θ
Greek capital letter iota	C9	Ι
Greek capital letter kappa	CA	Κ
Greek capital letter lambda	CB	Λ
Greek capital letter mu	CC	Μ
Greek capital letter nu	CD	Ν
Greek capital letter xi	CE	Ξ
Greek capital letter omicron	CF	Ο
Greek capital letter pi	D0	Π
Greek capital letter rho	D1	Ρ
Greek capital letter sigma	D3	Σ
Greek capital letter tau	D4	Τ
Greek capital letter upsilon	D5	Υ
Greek capital letter phi	D6	Φ
Greek capital letter chi	D7	Χ
Greek capital letter psi	D8	Ψ
Greek capital letter omega	D9	Ω

10 APPENDIX C: CHANGELOG

Date	Description
26/10/2016	Added new Optional Parameter P for delivery notifications.
13/10/2016	Added internal status code 451A for Optional Parameters.
09/05/2016	Subject Field limitation increased to 255 characters Additional Optional Parameters for Delivery Notifications updated
22/03/2016	SplitText parameter is now default to YES.
02/11/2015	Added information on optional parameter for delivery notifications. Corrected some examples
26/05/2015	Added information on optional parameters for delivery notifications. New message example: UCS2 emoji