ETSI TS 129 199-4 V6.3.0 (2005-06)

Technical Specification

Universal Mobile Telecommunications System (UMTS); Open Service Access (OSA); Parlay X web services; Part 4: Short messaging (3GPP TS 29.199-04 version 6.3.0 Release 6)

Reference

RTS/TSGC-0529199-04v630

Keywords

UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00   Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

ETSI

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp .

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

3GPP acknowledges the contribution of the Parlay X Web Services specifications from The Parlay Group. The Parlay Group is pleased to see 3GPP acknowledge and publish the present document, and the Parlay Group looks forward to working with the 3GPP community to improve future versions of the present document.

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

The present document is part 4 of a multi-part deliverable covering the 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Open Service Access (OSA); Parlay X Web Services, as identified below:

Part 1:   "Common";
Part 2:   "Third party call";
Part 3:   "Call Notification";
Part 4:   "Short Messaging";
Part 5:   "Multimedia Messaging";
Part 6:   "Payment";
Part 7:   "Account management";
Part 8:   "Terminal Status";
Part 9:   "Terminal location";
Part 10:   "Call handling";
Part 11:   "Audio call";
Part 12:   "Multimedia conference";
Part 13:   "Address list management";
Part 14:   "Presence".

1 Scope

The present document is Part 4 of the Stage 3 Parlay X Web Services specification for Open Service Access (OSA).

The OSA specifications define an architecture that enables application developers to make use of network functionality through an open standardized interface, i.e. the OSA APIs. The concepts and the functional architecture for the OSA are contained in 3GPP TS 23.198 [3]. The requirements for OSA are contained in 3GPP TS 22.127 [2].

The present document specifies the Short Messaging Web Service aspects of the interface. All aspects of the Short Messaging Web Service are defined here, these being:

- Name spaces.
- Sequence diagrams.
- Data definitions.
- Interface specification plus detailed method descriptions.
- Fault definitions.
- Service policies.
- WSDL description of the interfaces.

The present document has been defined jointly between 3GPP TSG CT WG5, ETSI TISPAN and The Parlay Group.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1]   3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2]   3GPP TS 22.127: "Service Requirement for the Open Services Access (OSA); Stage 1".

[3]   3GPP TS 23.198: "Open Service Access (OSA); Stage 2".

[4]   3GPP TS 22.101: "Service aspects; Service principles".

[5]   W3C Recommendation (2 May 2001): "XML Schema Part 2: Datatypes".

NOTE: Available at http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/.

[6]   3GPP TS 29.199-1: "Open Service Access (OSA); Parlay X Web Services; Part 1: Common".

[7]   3GPP TS 23.040: "Technical realization of Short Message Service (SMS)".

[8]   RFC2822: "Internet Message Format".

NOTE: Available at htpp://www.ietf.org/rfc/rfc2822.txt

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 29.199-1 [6] apply.

Additionlly the following definition is needed:

Shortcode: a short telephone number, usually 4 to 6 digits long. This is represented by the 'tel:' URI defined in [6].

Whitespace: see definition for CFWS as defined in RFC2822 [8].

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TS 29.199-1 [6] and the following apply:

    SMS   Short Message Service
    SMS-C   Short Message Service - Center

4 Detailed service description

Currently, in order to programmatically receive and send SMS it is necessary to write applications using specific protocols to access SMS functions provided by network elements (e.g. SMS-C). This approach requires a high degree of network expertise. Alternatively it is possible to use the Parlay/OSA approach, invoking standard interfaces (e.g. User Interaction or Messaging Service Interfaces) to gain access to SMS capabilities, but these interfaces are usually perceived to be quite complex by IT application developers. Developers must have advanced telecommunication skills to use OSA interfaces.

In this clause is described a Parlay X Web Service, for sending and receiving SMS messages. The overall scope of this Web Service is to provide to application developers primitives to handle SMS in a simple way. In fact, using the SMS Web Service, application developers can invoke SMS functions without specific Telco knowledge.

ShortMessaging provides operations (see clause 8.1, Send SMS API) for sending a SMS message to the network and a polling mechanism for monitoring the delivery status of a sent SMS message. It also provides an asynchronous notification mechanism for delivery status (see clause 8.2, SmsNotification API).

ShortMessaging also allows an application to receive SMS messages. Both a polling (see clause 8.3, ReceiveSMS API) and an asynchronous notification mechanism (see clause 8.2, SMSNotification API and clause 8.4, SmsNotificationManager API) are available.

Figure 1 shows a scenario using the SMS Web Service to send an SMS message from an application. The application invokes a Web Service to retrieve a weather forecast for a subscriber (1) and (2) and a Parlay X Interface (3) to use the SMS Web Service operations (i.e. to send an SMS). After invocation, the SMS Web Service invokes a Parlay API method (4) using the Parlay/OSA SCS (Generic User Interaction) interface. This SCS handles the invocation and sends an UCP operation (5) to an SMS-C. Subsequently the weather forecast is delivered (6) to the subscriber.

In an alternative scenario, the Parlay API interaction involving steps (4) and (5) could be replaced with a direct interaction between the SMS Web Service and the Mobile network.
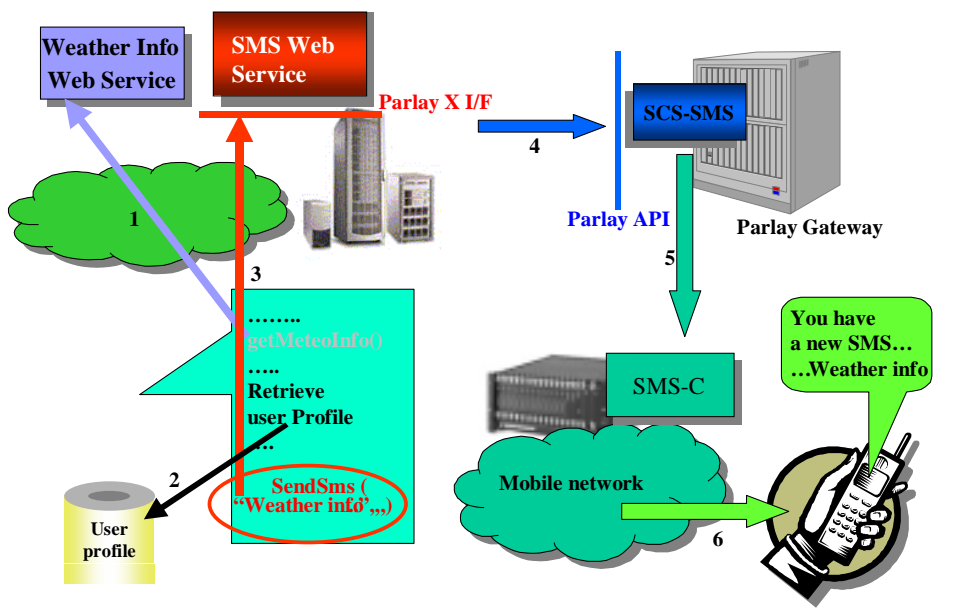
Figure 1: Send SMS Scenario

Figure 2 shows a scenario using the SMS Web Service to deliver a received SMS message to an application. The application receives a Parlay X Web Service invocation for an SMS sent by a subscriber (1) and (2). The SMS message contains the e-mail address of the person the user wishes to call. The application invokes a Parlay X Interface (3) to the Third Party Call Web Service in order to initiate the call (4).
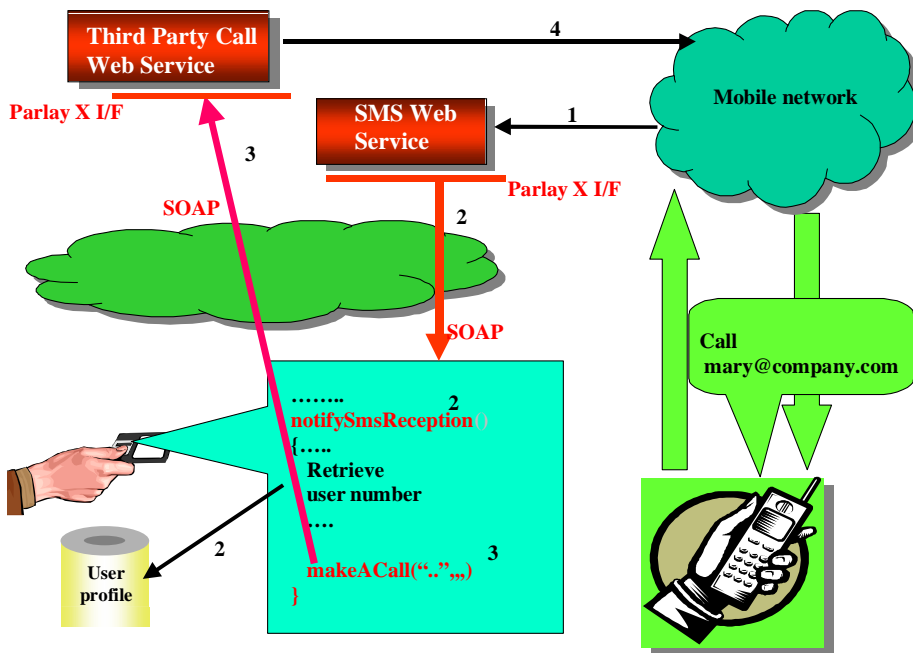
Figure 2: Receive SMS Scenario

5 Namespaces

The SendSms interface uses the namespace:

   http://www.csapi.org/wsdl/parlayx/sms/send/v2_1

The ReceiveSms interface uses the namespace:

   http://www.csapi.org/wsdl/parlayx/sms/receive/v2_1

The SmsNotification interface uses the namespace:

   http://www.csapi.org/wsdl/parlayx/sms/notification/v2_1

The SmsNotificationManager interface uses the namespace:

   http://www.csapi.org/wsdl/parlayx/sms/notification_manager/v2_2

The data types are defined in the namespace:

   http://www.csapi.org/schema/parlayx/sms/v2_1

The 'xsd' namespace is used in the present document to refer to the XML Schema data types defined in XML Schema [5]. The use of the name 'xsd' is not semantically significant.

6 Sequence Diagrams

6.1 Send SMS and report status

Sending SMS message from Web portals is a common capability offered by Service Providers. This sequence diagram shows a portal providing this service.
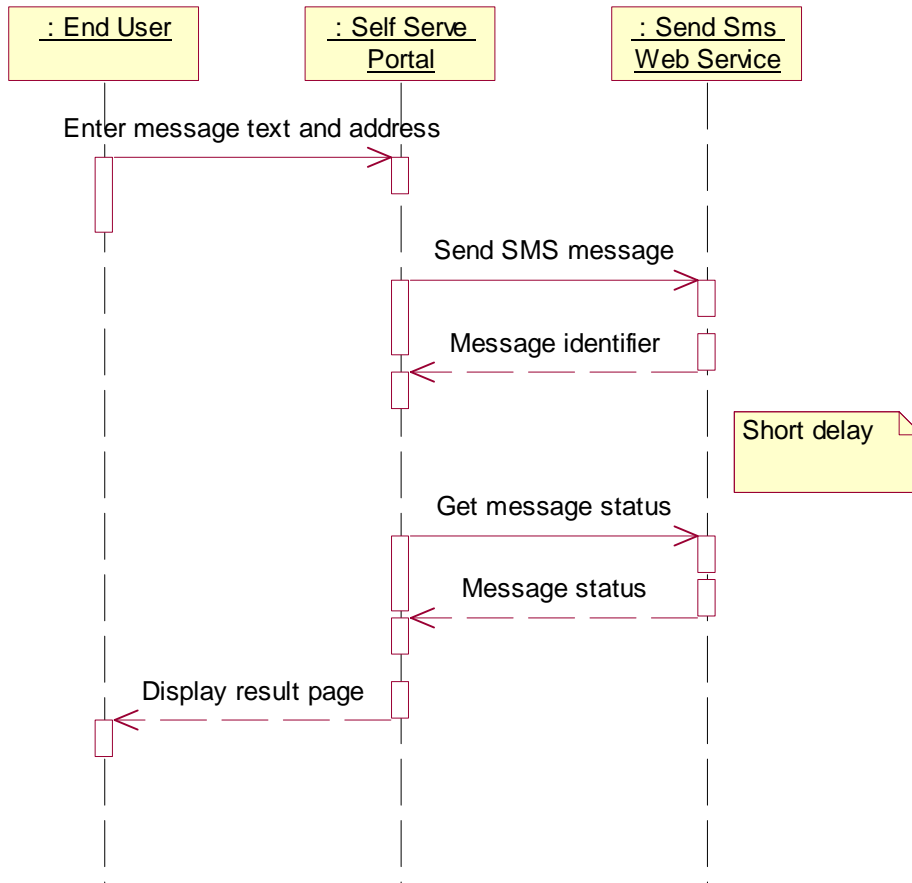
Figure 3

7 XML Schema data type definition

7.1 DeliveryStatus enumeration

List of delivery status values.

| Enumeration	Description
DeliveredToNetwork	Successful delivery to network
DeliveryUncertain	Delivery status unknown: e.g. because it was handed off to another network.
DeliveryImpossible	Unsuccessful delivery; the message could not be delivered before it expired.
MessageWaiting	The message is still queued for delivery. This is a temporary state, pending transition to one of the preceding states.
DeliveredToTerminal	Successful delivered to Terminal
DeliveryNotificationNotSupported	Unable to provide delivery receipt notification. NotifySMSDeliveryReceipt function will provide 'DeliveryNotificationNotSupported' to indicate that delivery receipt for the specified address in a SendSMSRequest is not supported.

7.2 SmsFormat enumeration

List of SMS format values.

Enumeration	Description
Ems	Enhanced Messaging Service, standardized in 3GPP TS 23.040 [7], which defines a logo/ringtone format
SmartMessaging[TM]	Defines a logo/ringtone format

7.3 DeliveryInformation structure

Delivery status information.

Element name	Element type	Optional	Description
Address	xsd:anyURI	No	It indicates the destination address to which the notification is related.
DeliveryStatus	DeliveryStatus	No	Indicates the delivery result for destinationAddress. Possible values are: • 'Delivered'; • 'DeliveryUncertain'; • 'DeliveryImpossible'.

7.4 SmsMessage structure

SMS message information. The SenderAddress is the address from which the message was actually sent, which may or may not match the senderName value provided in the SendSms operation.

Element name	Element type	Optional	Description
Message	xsd:string	No	Text received in SMS
SenderAddress	xsd:anyURI	No	It indicates address sending the SMS
SmsServiceActivation Number	xsd:anyURI	No	Number associated with the invoked Message service, i.e. the destination address used to send the message

8 Web Service interface definition

8.1 Interface: SendSms

8.1.1 Operation: SendSms

The invocation of **sendSms** requests to send an SMS, specified by the String **Message** to the specified address (or address set), specified by **Addresses**. Optionally the application can also indicate the sender name (**SenderName**), i.e. the string that is displayed on the user's terminal as the originator of the message, the charging information and a ReceiptRequest. The ReceiptRequest which is a SimpleReference structure indicates the application endpoint, interface used for notification of delivery receipt and a correlator that uniquely identifies the sending request. By invoking this operation with the optional ReceiptRequest parameter the application requires to receive the notification of the status of the SMS delivery.

If Notification mechanism is not supported by a network a fault (SVC0283) will be returned to the application and the message will not be sent to the addresses specified. Notification to the application is done by invoking the **notifiySMSDeliveryReceipt** operation at the endpoint specified in ReceiptRequest.

The application can also explicitly invoke the **getSmsDeliveryStatus** using the **RequestIdentifier** returned by the **sendSMS** invocation to get the delivery status.

Addresses may include group URIs as defined in the Address List Management specification. If groups are not supported, a PolicyException (POL0006) will be returned to the application.

For GSM systems, if **Message** contains characters not in the GSM 7-bit character set, the SMS is sent as a Unicode SMS.

If **Message** is longer than the maximum supported length (e.g. for GSM, 160 GSM 7-bit characters or 70 Unicode characters), the message will be sent as several concatenated short messages.

The correlator provided in the ReceiptRequest must be unique for this Web Service and application at the time the notification is initiated, otherwise a ServiceException (SVC0005) will be returned to the application.

8.1.1.1 Input message: SendSmsRequest

Part name	Part type	Optional	Description
Addresses	xsd:anyURI [1..unbounded]	No	Addresses to which the SMS will be sent
SenderName	xsd:string	Yes	If present, it indicates the SMS sender name, i.e. the string that is displayed on the user's terminal as the originator of the message
Charging	common:ChargingInformation	Yes	Charge to apply to this message
Message	xsd:string	No	Text to be sent in SMS
ReceiptRequest	common:SimpleReference	Yes	It defines the application endpoint, interfaceName and correlator that will be used to notify the application when the message has been delivered to terminal or if delivery is impossible.

8.1.1.2 Output message : SendSmsResponse

Part name	Part type	Optional	Description
RequestIdentifier	xsd:string	No	It identifies a specific SMS delivery request

8.1.1.3 Referenced faults

ServiceException from [6]:

- SVC0001 - Service error.

- SVC0002 - Invalid input value.

- SVC0004 - No valid addresses.

- SVC0006 - Invalid group.

- SVC0280 - Message too long.

- SVC0283 – Delivery Receipt Notification not supported

PolicyException from [6]:

- POL0001 - Policy error.

- POL0006 - Groups not allowed.

- POL0007 - Nested groups not allowed.

- POL0008 - Charging not allowed.

8.1.2 Operation: SendSmsLogo

The invocation of **sendSmsLogo** requests to send an SMS logo, specified by the byte array **image** to the specified address (or address set), specified by **destinationAddressSet**. Optionally the application can also indicate the sender name (**senderName**), i.e. the string that is displayed on the user's terminal as the originator of the message, the charging information (**charging**) and a ReceiptRequest. The **receiptRequest** which is a SimpleReference structure indicates the application endpoint, interface used for notification of delivery receipt and a correlator that uniquely identifies the sending request. By invoking this operation with the optional **receiptRequest** parameter the application requires to receive the notification of the status of the SMS delivery.

If Notification mechanism is not supported by a network a serviceexception(SVC0283) will be returned to the application and the message will not be sent to the addresses specified. Notification to the application is done by invoking the **notifiySMSDeliveryReceipt** operation at the endpoint specified in ReceiptRequest.

The application can also explicitly invoke the **getSmsDeliveryStatus** using the **requestIdentifier** returned by the **sendSMSLogo** invocation to get the delivery status.

Addresses may include group URIs as defined in the Address List Management specification. If groups are not supported, a PolicyException (POL0006) will be returned to the application.

The correlator provided in the ReceiptRequest must be unique for this Web Service and application at the time the notification is initiated, otherwise a ServiceException (SVC0005) will be returned to the application.

8.1.2.1 Input message: SendSmsLogoRequest

Part name	Part type	Optional	Description
Addresses	xsd:anyURI [1..unbounded]	No	Addresses to which the SMS logo will be sent
SenderName	xsd:string	Yes	SMS sender name, i.e. the string that is displayed on the user's terminal as the originator of the message
Charging	common:ChargingInformation	Yes	Charge to apply to this message
Image	xsd:base64Binary	No	The image in jpeg, gif or png format. The image will be scaled to the proper format
SmsFormat	SmsFormat	No	Possible values are: 'Ems' or 'SmartMessaging'
ReceiptRequest	common:SimpleReference	Yes	It defines the application endpoint, interfaceName and correlator that will be used to notify the application when the message has been delivered to terminal or if delivery is impossible

8.1.2.2 Output message: SendSmsLogoResponse

Part name	Part type	Optional	Description
requestIdentifier	String	No	It identifies a specific SMS delivery request

8.1.2.3 Referenced faults

ServiceException from [6]:

- SVC0001 - Service error.

- SVC0002 - Invalid input value.

- SVC0004 - No valid addresses.

- SVC0006 - Invalid group.

- SVC0281 - Unrecognized data format.

- SVC0283 – Delivery Receipt Notification not supported

PolicyException from [6]:

- POL0001 - Policy error.

- POL0006 - Groups not allowed.

- POL0007 - Nested groups not allowed.

- POL0008 - Charging not allowed.

8.1.3 Operation: SendSmsRingtone

The invocation of **sendSmsRingtone** requests to send an SMS ringtone, specified by the String **ringtone** (in RTX format) to the specified addresses, specified by **Addresses**. Optionally the application can also indicate the sender name (**senderName**) i.e. the string that is displayed on the user's terminal as the originator of the message, the charging information (**charging**) and a **receiptRequest**. The **receiptRequest** which is a SimpleReference structure indicates the application endpoint, interface used for notification of delivery receipt and a correlator that uniquely identifies the sending request. By invoking this operation with the optional **receiptRequest** parameter the application requires to receive the notification of the status of the SMS delivery.

If Notification mechanism is not supported by a network a fault (SVC0283) will be returned to the application and the message will not be sent to the addresses specified. Notification to the application is done by invoking the **notifiySMSDeliveryReceipt** operation at the endpoint specified in ReceiptRequest.

The application can also explicitly invoke the **getSmsDeliveryStatus** using the **requestIdentifier** returned by the **sendSMSRingTone** invocation to get delivery status..

Addresses may include group URIs as defined in the Address List Management specification. If groups are not supported, a PolicyException (POL0006) will be returned to the application.

The correlator provided in the ReceiptRequest must be unique for this Web Service and application at the time the notification is initiated, otherwise a ServiceException (SVC0005) will be returned to the application.

Depending on the length of the ringtone, it may be sent as several concatenated short messages.

    NOTE: On the RTX Ringtone Specification : An RTX file is a text file, containing the ringtone name, a control subclause and a subclause containing a comma separated sequence of ring tone commands.

8.1.3.1   Input message: SendSmsRingtoneRequest

Part name	Part type	Optional	Description
Addresses	xsd:anyURI [1..unbounded]	No	Addresses to which the SMS logo will be sent
SenderName	xsd:string	Yes	SMS sender name, i.e. the string that is displayed on the user's terminal as the originator of the message
Charging	common:ChargingInformation	Yes	Charge to apply to this message
Ringtone	xsd:string	No	The ringtone in RTX format (see note above). (http://www.logomanager.co.uk/help/Edit/RTX.html)
SmsFormat	SmsFormat	No	Possible values are: 'Ems' or 'SmartMessaging'
ReceiptRequest	common:SimpleReference	Yes	It defines the application endpoint, interfaceName and correlator that will be used to notify the application when the message has been delivered to terminal or if delivery is impossible

8.1.3.2   Output message: SendSmsRingtoneResponse

Part name	Part type	Optional	Description
RequestIdentifier	xsd:string	No	It identifies a specific SMS delivery request

8.1.3.3   Referenced faults

ServiceException from [6]:

- SVC0001 - Service error.

- SVC0002 - Invalid input value.

- SVC0004 - No valid addresses.

- SVC0006 - Invalid group.

- SVC0281 - Unrecognized data format.

- SVC0283 – Delivery Receipt Notification not supported

PolicyException from [6]:

- POL0001 - Policy error.

- POL0006 - Groups not allowed.

- POL0007 - Nested groups not allowed.

- POL0008 - Charging not allowed.

8.1.4   Operation: GetSmsDeliveryStatus

The invocation of **getSmsDeliveryStatus** requests the status of a previous SMS delivery request identified by **requestIdentifier**. The information on the status is returned in **deliveryStatus**, which is an array of status related to the request identified by **requestIdentifier**. The status is identified by a couplet indicating a user address and the associated delivery status. This method can be invoked multiple times by the application even if the status has reached a final value. However, after the status has reached a final value, status information will be available only for a limited period of time that should be specified in an off-line configuration step. The following four different SMS delivery status have been identified:

- 'DeliveredToNetwork': in case of concatenated messages, only when all the SMS-parts have been successfully delivered to the network.

- 'DeliveryUncertain': e.g. because it was handed off to another network.

- 'DeliveryImpossible': unsuccessful delivery; the message could not be delivered before it expired.

- 'MessageWaiting': the message is still queued for delivery.

- 'DeliveredToTerminal': in case of concatenated messages, only when all the SMS-parts have been successfully delivered to the terminal.

8.1.4.1 Input message: GetSmsDeliveryStatusRequest

Part name	Part type	Optional	Description
RequestIdentifier	xsd:string	No	It identifies a specific SMS delivery request

8.1.4.2 Output message : GetSmsDeliveryStatusResponse

Part name	Part type	Optional	Description
DeliveryStatus	DeliveryInformation [0..unbounded]	Yes	It lists the variations on the delivery status of the SMS

8.1.4.3 Referenced faults

ServiceException from [6]:

- SVC0001 - Service error.

- SVC0002 - Invalid input value.

PolicyException from [6]:

- POL0001 - Policy error.

8.2 Interface: SmsNotification

SmsNotification is the application side notification interface to which short messages are delivered.

8.2.1 Operation: NotifySmsReception

The notification is used to send a short message to the application. The notification will occur only if the SMS fulfils the criteria specified when starting the SMS notification.

The **notifySmsReception** method must be implemented by a Web Service at the *application side*. It will be invoked by the Parlay X server to notify the application of the reception of an SMS. The notification will occur if and only if the SMS received fulfils the criteria specified in a provisioning step, identified by the **correlator**. The criteria must at least include an **smsServiceActivationNumber**, i.e. the SMS destination address that can be "monitored" by the application. The parameter **senderAddress** contains the address of the sender. The application can apply the appropriate service logic to process the SMS.

8.2.1.1 Input message: NotifySmsReceptionRequest

Part name	Part type	Optional	Description
correlator	xsd:string	No	Correlator provided in request to set up this notification
Message	SmsMessage	No	Message received

8.2.1.2 Output message : NotifySmsReceptionResponse

Part name	Part type	Optional	Description
None			

8.2.1.3 Referenced faults

None.

8.2.2 Operation: NotifySmsDeliveryReceipt

The **notifySmsDeliveryReceipt** method must be implemented by a Web Service at the *application side* if it requires notification of SMSdelivery receipt. It will be invoked by the Parlay X server to notify the application when a SMS sent by an application has been delivered to the terminal of the recipient or if delivery is impossible. The notification will occur if and only if the status of the sent SMS is "DeliveredToTerminal" or "DeliveryImpossible" and the application has specified interest in notification when sending an SMS message by specifying the optional receiptRequest parameter. The correlator returned corresponds to the identifier specified by the application in the **receiptRequest** of the original **sendSMS** request

When a SMS message is sent to multiple addresses, the notification from the server will send notification for each terminal as and when a SMS message is delivered to a terminal.

The following three different SMS delivery status will be returned in NotifySMSDeliveryReceiptResponse:

- 'DeliveryImpossible': unsuccessful delivery; the message could not be delivered before it expired.

- 'DeliveredToTerminal': in case of concatenated messages, only when all the SMS-parts have been successfully delivered to the terminal.

- "DeliveredNotificationNotSupported" - If notification is supported by the network but it does not support delivery receipt for one or more addresses specified in the **sendSMS** message. The service will send this status for those addresses.

8.2.2.1 Input message: NotifySmsDeliveryReceiptRequest

Part name	Part type	Optional	Description
Correlator	xsd:string	No	The identifier defining the original SendRequest. This correlator was passed by the application during the SendSMS request
DeliveryStatus	DeliveryInformation	No	It lists the variations on the delivery status of the SMS to a terminal

8.2.2.2 Output message: NotifySmsDeliveryReceiptResponse

Part name	Part type	Optional	Description
None			

8.2.2.3 Referenced faults

None.

8.3 Interface: ReceiveSms

8.3.1 Operation: GetReceivedSms

The invocation of **getReceivedSms** retrieves all the SMS messages received that fulfil the criteria identified by **registrationIdentifier**. The method returns only the list of SMS messages received since the previous invocation of the same method, i.e. each time the method is executed the messages returned are removed from the server. Moreover, each SMS message will be automatically removed from the server after a maximum time interval specified in an off-line configuration step.

The received SMS messages are returned in **receivedSms**. An SMS message is identified by a structure indicating the sender of the SMS message and the content.

8.3.1.1   Input message: GetReceivedSmsRequest

Part name	Part type	Optional	Description
RegistrationIdentifier	xsd:string	No	Identifies the off-line provisioning step that enables the application to receive notification of SMS reception according to specified criteria

8.3.1.2   Output message : GetReceivedSmsResponse

Part name	Part type	Optional	Description
ReceivedSms	SmsMessage [0..unbounded]	Yes	It lists the received SMS since last invocation

8.3.1.3   Referenced faults

ServiceException from [6]:

- SVC0001 - Service error.

- SVC0002 - Invalid input value.

PolicyException from [6]:

- POL0001 - Policy error.

8.4   Interface: SmsNotificationManager

The short message notification manager enables applications to set up and tear down notifications for short messages, online. The means to provision notifications offline are not specified here.

8.4.1   Operation: StartSmsNotification

Start notifications to the application for a given SMS Service activation number and criteria.

The SMS Service activation number is an Address Data item as defined in 3GPP TS 29.199-1 [6]. A Shortcode is an example of an Address Data item.

The correlator provided in the reference must be unique for the application Web Service at the time the notification is initiated, otherwise a ServiceException (SVC0005) will be returned to the application..

If specified, criteria will be used to filter messages that are to be delivered to an application. If criteria are not provided, or is an empty string, then all messages for the SmsServiceActivationNumber will be delivered to the application. The SmsServiceActivationNumber and criteria combination must be unique. If a criteria overlaps then SVC0008 will be returned to the application and the notification will not be set up. Note that the use of criteria will allow different notification endpoints to receive notifications for the same SmsServiceActivationNumber. The combination of SmsServiceActivationNumber and criteria must be unique, so that a notification will be delivered to only one notification endpoint. If no match is found, the message will not be delivered to the application.

8.4.1.1 Input message: StartSmsNotificationRequest

Part name	Part type	Optional	Description
Reference	common:SimpleReference	No	Notification endpoint definition
SmsServiceActivation Number	xsd:anyURI	No	the destination address to the short message
Criteria	xsd:string	Yes	The text to match against to determine the application to receive the notification. This text is matched against the first word in the message, defined as the initial characters after discarding any leading whitspaceWhitespace and ending with a whitespaceWhitespace or end of message. The matching shall be case-insensitive.

8.4.1.2 Output message: StartSmsNotificationResponse

Part Name	Part Type	Optional	Description
none			

8.4.1.3 Referenced Faults

ServiceException from [6]

- SVC0001 – Service error

- SVC0002 – Invalid input value

- SVC0005 – Duplicate correlator

- SVC0008 – Overlapping Criteria

PolicyException from [6]

- POL0001 – Policy error

8.4.2 Operation: StopSmsNotification

The application may end a short message notification using this operation

8.4.2.1 Input message: StopSmsNotificationRequest

Part name	Part type	Optional	Description
Correlator	xsd:string	No	Correlator of request to end

8.4.2.2 Output message: StopSmsNotificationResponse

Part Name	Part Type	Optional	Description
None			

8.4.2.3 Referenced Faults

ServiceException from [6]

- SVC0001 – Service error

- SVC0002 – Invalid input value

PolicyException from [6]

- POL0001 – Policy error

9 Fault definitions

9.1 ServiceException

9.1.1 SVC0280: Message too long

Name	Description
Message Id	SVC0280
Text	Message too long. Maximum length is %1 characters
Variables	%1 Number of characters allowed in a message

9.1.2 SVC0281: Unrecognized data format

Name	Description
Message Id	SVC0281
Text	Data format not recognized for message part %1
Variables	%1 Message part with the unrecognized data

9.1.3 Void

The fault code (SVC0282) is reserved and shall not be used,

9.1.4 SVC0283: Delivery Receipt Notification not supported

Name	Description
Message Id	SVC0283
Text	Delivery Receipt Notification not supported
Variables	

10 Service policies

Service policies for this service.

Name	Type	Description
GroupSupport	xsd:boolean	Groups may be included with addresses
NestedGroupSupport	xsd:boolean	Are nested groups supported in group definitions
ChargingSupported	xsd:boolean	Is charging supported for send operations

Annex A (normative):
WSDL for Short Messaging

The document/literal WSDL representation of this interface specification is compliant to 3GPP TS 29.199-1 [6] and is contained in text files (contained in archive 29199-04-630-doclit.zip) which accompanies the present document.

Annex B (informative): Change history

					Change history		
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
Dec 2003	CN_21	NP-030552	--	--	Submitted to CN#22 for Information	1.0.0	

Jan 2004	--	--	--	--	Added The W3C WSDL representation of the APIs specified in the present document is contained in a set of files which accompany the present document:
					   px0326rpcenc.zip
					   px0326rpclit.zip
						1.0.1	

Jun 2004	CN_24	NP-040274	--	--	Split into multi-part specification. 29.199-0n, for n=1,2…9. Submitted to CN#24 for Information	1.0.3	
Sep 2004	CN_25	NP-040360	--	--	Draft v200 submitted to TSG CN#25 for Approval.	2.0.0	6.0.0
Dec 2004	CN_26	NP-040487	0001	--	Add SmsNotificationManager interface to PXWS Short-Messaging	6.0.0	6.1.0
Dec 2004	CN_26	NP-040609	0002	1	Add PXWS SMS Notification Delivery Reception	6.0.0	6.1.0
Mar 2005	CN_27	NP-050021	0003	--	Correct criteria	6.1.0	6.2.0
Mar 2005	CN_27	NP-050021	0004	--	Fix StopSMSNotification message part	6.1.0	6.2.0
Jun 2005	CT_28	CP-050221	0005	--	Optionals for Part 4	6.2.0	6.3.0

History

Document history		
V6.1.0	December 2004	Publication
V6.2.0	March 2005	Publication
V6.3.0	June 2005	Publication