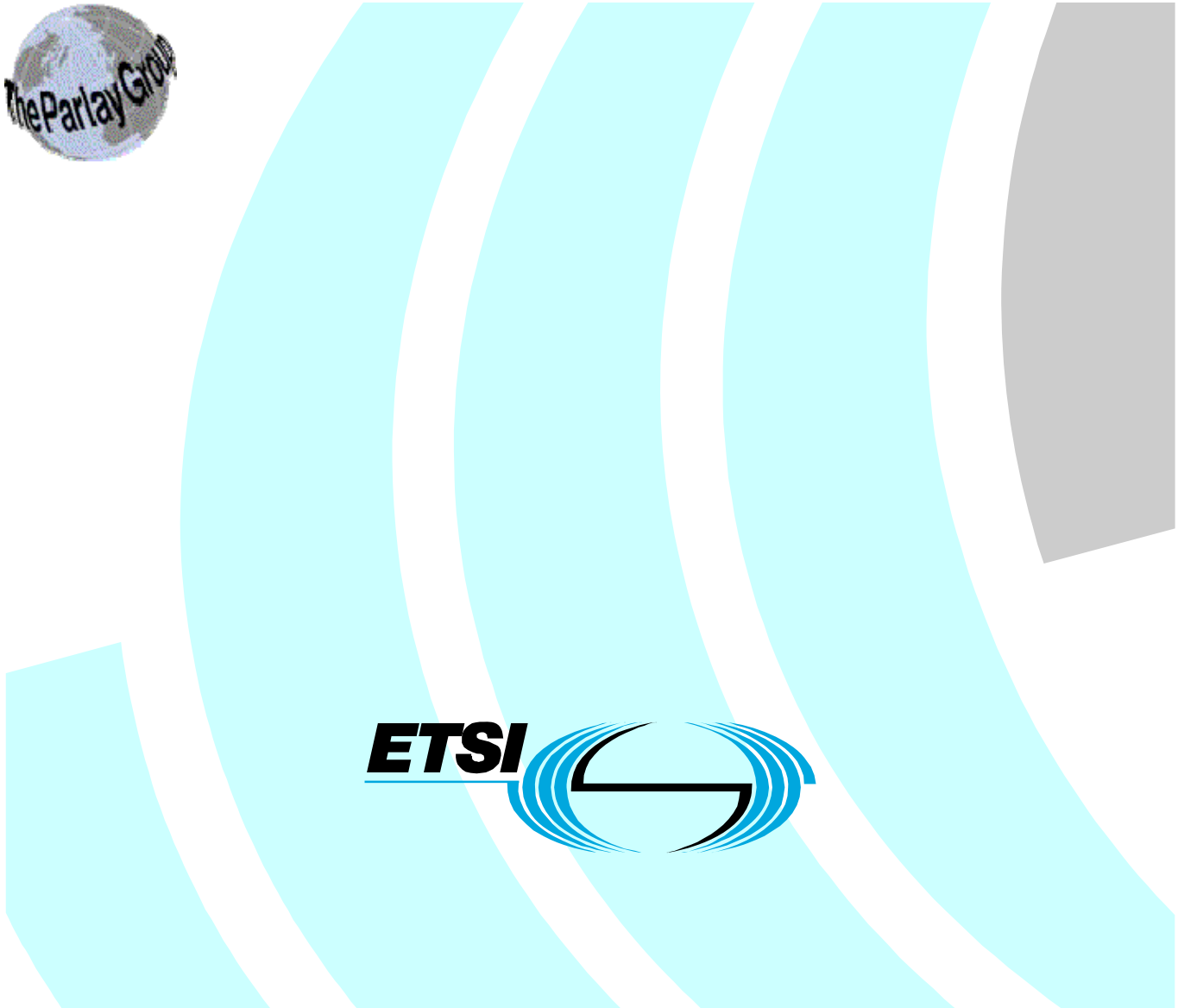
Final draft ETSI ES 202 504-15 V1.1.1 (2008-02)

ETSI Standard

Open Service Access (OSA); Parlay X Web Services; Part 15: Message Broadcast (Parlay X 3)



Reference
DES/TISPAN-01034-15-OSA

Keywords
API, OSA, service

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

Contents

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 ETSI Standard (ES) has been produced by ETSI Technical Committee Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN), and is now submitted for the ETSI standards Membership Approval Procedure.

The present document is part 15 of a multi-part deliverable covering Open Service Access (OSA); Parlay X 3 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";

Part 15: "Message Broadcast";

Part 16: "Geocoding";

Part 17: "Application-driven Quality of Service (QoS)";

Part 18: "Device Capabilities and Configuration";

Part 19: "Multimedia Streaming Control";

Part 20: "Multimedia Multicast Session Management".

The present document has been defined jointly between ETSI, The Parlay Group (http://www.parlay.org) and the 3GPP.

The present document forms part of the Parlay X 3.0 set of specifications.

The present document is equivalent to 3GPP TS 29.199-15 V7.2.0 (Release 7).

1 Scope

The present document is part 15 of the Stage 3 Parlay X 3 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 present document specifies the Message Broadcast Web Service. The following are defined here:

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

2 References

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

- For a specific reference, subsequent revisions do not apply.
- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:
  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;
  - for informative references.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

For online referenced documents, information sufficient to identify and locate the source shall be provided. Preferably, the primary source of the referenced document should be cited, in order to ensure traceability. Furthermore, the reference should, as far as possible, remain valid for the expected life of the document. The reference shall include the method of access to the referenced document and the full network address, with the same punctuation and use of upper case and lower case letters.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents are indispensable for the application of the present document. For dated references, only the edition cited applies. For non-specific references, the latest edition of the referenced document (including any amendments) applies.

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

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

[2] ETSI ES 202 504-1: "Open Service Access (OSA); Parlay X Web Services; Part 1: Common (Parlay X 3)".

[3] ETSI TS 123 041: "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Technical realization of Cell Broadcast Service (CBS) (3GPP TS 23.041)".

[4] ETSI TS 123 032: "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Universal Geographical Area Description (GAD) (3GPP TS 23.032)".

[5] ETSI ES 202 504-9: "Open Service Access (OSA); Parlay X Web Services; Part 9: Terminal Location (Parlay X 3)".

[6] ETSI ES 202 504-4: "Open Service Access (OSA); Parlay X Web Services; Part 4: Short Messaging (Parlay X 3)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in ES 202 504-1 [2] apply.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in ES 202 504-1 [2], ES 202 504-4 [6] and the following apply.

CBC	Cell Broadcast Centre
CBS	Cell Broadcast Service

4 Detailed service description

Message broadcast is functionality that allows an application to send messages to all the fixed or mobile terminals in a specified geographical area.

Message broadcast provides operations for sending a broadcast message to the network and a polling mechanism for monitoring the delivery status of a sent broadcast message. It also provides an asynchronous notification mechanism for broadcast delivery status. In addition, a mechanism is provided to start and stop the notification of delivery receipts.

There are various use cases of using Message Broadcast Web Service including the commercial application. This Web Service could be also used for non-commercial purposes as follows:

- To provide area-based public information such as weather, traffic and other information of common interest.
- To provide emergency information such as severe weather warnings (e.g. typhoon, tsunami), environmental hazards (e.g. chemical spills) and terrorism alerts.

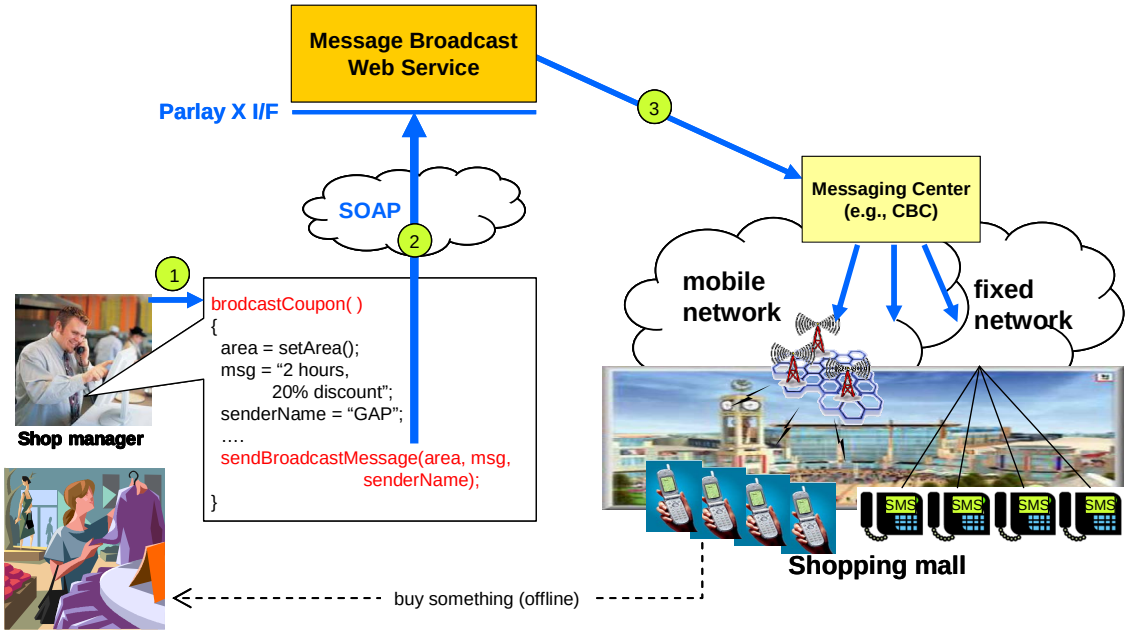


Figure 1: Send Broadcast Message Scenario

Figure 1 shows an advertising scenario using the Message Broadcast Web Service to broadcast messages describing shop discount offers inside, and in the vicinity of, a shopping mall. A shop manager who wants to increase sales during a holiday period can make use of a message broadcast application. By using the application, the manager can set the targeted area, compose the sales message and identify the shop offering the discount (1). Then, the application uses the Parlay X interface to invoke the Message Broadcast Web Service operation (2). After invocation, the Message Broadcast Web Service sends a message delivery operation to the messaging centre, e.g. the CBC (3). Subsequently, the shop discount message is delivered to all the terminals within the targeted area.

5 Namespaces

The SendBroadcastMessage interface uses the namespace:

http://www.csapi.org/wsdl/parlayx/message_broadcast/send/v3_1

The MessageBroadcastNotification interface uses the namespace:

http://www.csapi.org/wsdl/parlayx/message_broadcast/notification/v3_1

The MessageBroadcastNotificationManager interface uses the namespace:

http://www.csapi.org/wsdl/parlayx/message_broadcast/notification_manager/v3_0

The data types are defined in the namespace:

http://www.csapi.org/schema/parlayx/message_broadcast/v3_1

The 'xsd' namespace is used in the present document to refer to the XML Schema data types defined in XML Schema [1]. The use of the name 'xsd' is not semantically significant.

6 Sequence diagrams

6.1 Send broadcast message, get the status and cancel it

Pattern: Request/Response, One way

An application can send a broadcast message to a specific area and also poll for the delivery status from the Message Broadcast Web Service. If the application subsequently wishes to abort message broadcasting, it can send a cancellation request.

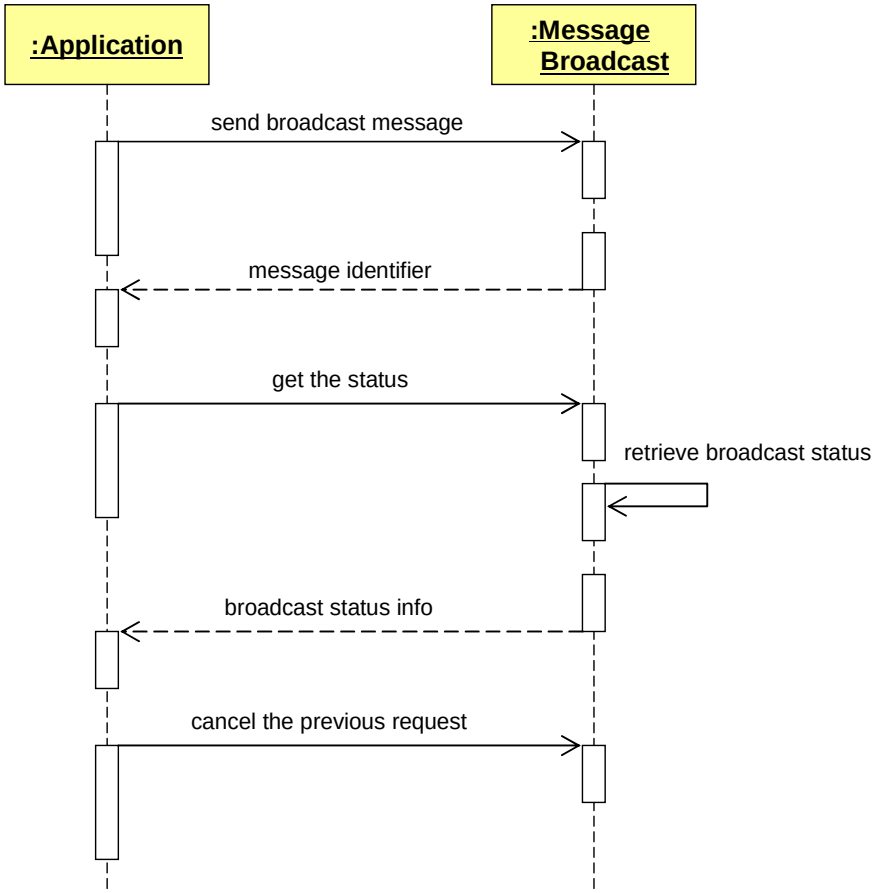


Figure 2: Message Broadcast Operations

6.2 Broadcast Status Notification

Pattern: Application Correlated Multiple Notification

An application can request a message be broadcast to multiple areas and, using a specified correlator parameter, can subsequently be notified of the broadcast delivery status for each area.

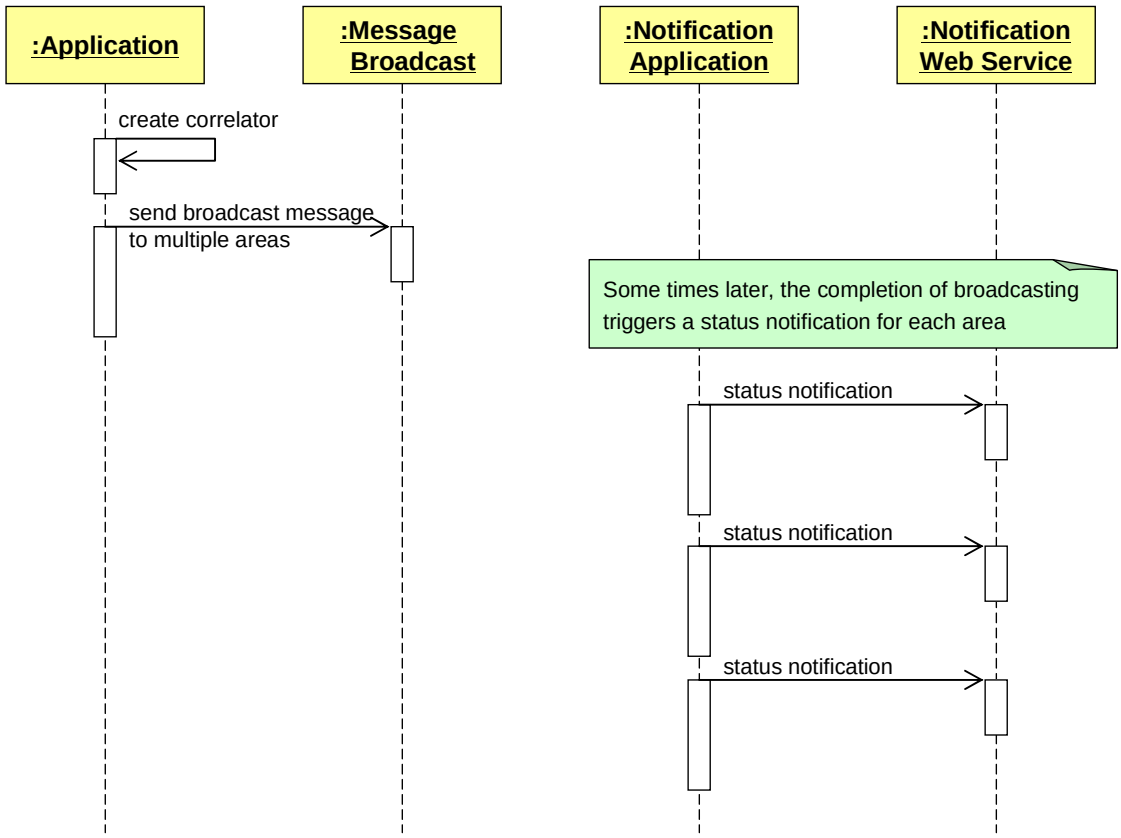


Figure 3: Message Broadcast Status Notification

7 XML Schema data type definition

7.1 BroadcastStatus Enumeration

List of possible broadcast delivery status values.

Enumeration value	Description
MessageWaiting	The message is still queued and not delivered to the network yet. Broadcasting has not commenced.
Broadcasting	Broadcasting is initiated and the network is still attempting to deliver messages: i.e. as many times as requested in the **totalBroadcasts** part of the **sendBroadcastMessageRequest** message.
Broadcasted	A final state that indicates broadcast requests were successfully delivered to network: i.e. as many times as requested.
BroadcastImpossible	Delivery of broadcast message is impossible. Reasons include: “out of network coverage”, “network overloads”, “expiry of valid period”.
BroadcastUnknown	Delivery status unknown: e.g. delivery requested but no response.
BroadcastNotificationNotSupported	Unable to provide broadcast delivery receipt notification. The **notifyBroadcastDeliveryReceipt** operation will return “BroadcastNotificationNotSupported” to indicate that delivery receipt for the specified area in a **sendBroadcastMessageRequest** message is not supported.

7.2 RetrievalStatus Enumeration

Enumeration of the status items that are related to an individual retrieval in a set, as described in clause 7.4.

Enumeration value	Description
Retrieved	Status retrieved, with result in **currentStatus** element.
NotRetrieved	Status not retrieved, **currentStatus** is not provided (does not indicate an error, no attempt may have been made).
Error	Error retrieving status.

7.3 BroadcastStatusInformation Structure

This indicates a broadcast status information of an area. It includes a mandatory broadcast status value with optional values to provide additional information such as the number of broadcasts, success rate, broadcast end time.

Element name	Element type	Optional	Description
status	BroadcastStatus	No	Broadcast status of this area.
numberOfBroadcasts	xsd:int	Yes	The number of broadcasts successfully sent out. This is optional and present only if **status** is either **Broadcasting** or **Broadcasted**.
successRate	xsd:int	Yes	Successful delivery rate expressed as a percentage. This is optional and present only if **status** is either **Broadcasting** or **Broadcasted**.
broadcastEndTime	xsd:dateTime	Yes	Completed time of broadcast. This is optional and present only if **status** is **Broadcasted**.

7.4 BroadcastStatusData Structure

Data structure containing area and its status. As this can be related to a query of multiple areas, the **reportStatus** element is used to indicate whether the status information for an area was retrieved or not, or if an error occurred.

Element name	Element type	Optional	Description
area	BroadcastArea	No	Broadcast area to which status information applies.
reportStatus	RetrievalStatus	No	Status of retrieval for this broadcast area.
currentStatus	BroadcastStatusInformation	Yes	Broadcast status of this area. It is only provided if **reportStatus=Retrieved**.
errorInformation	common:ServiceError	Yes	If **reportStatus** is **Error**, this is the reason for the error.

7.5 LocationPoint Structure

This is used to describe a location point. The definition of latitude and longitude values follows the terms defined in clause 7.1 of ES 202 504-9 [5]**.**

Element name	Element type	Optional	Description
latitude	xsd:float	No	Latitude value of a location
longitude	xsd:float	No	Longitude value of a location

7.6 Circle Structure

Circle representation of a geographical area.

Element name	Element type	Optional	Description
center	LocationPoint	No	The center point of circle
radius	xsd:float	No	Radius of circle (in meters)

7.7 Polygon Structure

Set of coordinates to configure the polygonal type of a geographical area.

Element name	Element type	Optional	Description
locationPoints	LocationPoint [3..13]	No	Set of location points to make a polygon. See also clause 5.7, 7.3.7 of TS 123 032 [4].

7.8 AreaType Enumeration

This indicates the types of area that may be used by an application to define a broadcast area.

Enumeration value	Description
Alias	Alias name shared by both application and network
Circle	Area represented as a circle shape
Polygon	Area represented as a polygon shape

7.9 BroadcastArea Union

Representation methods of broadcast area.

Element name	Element type	Optional	Description
UnionElement	AreaType	No	Type of geographical area. One of the following:
Alias	xsd:string	Yes	An alias name for a geographical area. The alias name shall be understood and translated by the network.
Circle	Circle	Yes	Circle representation
Polygon	Polygon	Yes	Polygon representation

7.10 MessagePriority Enumeration

List of delivery priority values.

Enumeration value	Description
Default	Default message priority
Low	Low message priority
Normal	Normal message priority
High	High message priority

8 Web Service interface definition

8.1 Interface: SendBroadcastMessage

This interface defines operations to send a broadcast message and to subsequently poll for delivery status. It also defines an operation to cancel the previous request.

8.1.1 Operation: sendBroadcastMessage

The application invokes the **sendBroadcastMessage** operation to send a broadcast message into the designated area(s), respectively specified by the mandatory **message** and **broadcastArea** parts:

- If **message** is longer than the maximum supported length in the network, the message content will be sent as several concatenated messages. For example a CBS message can consist of up to 15 CBS pages, where each page contains 82 octets, which, using the default character set, equates to 93 characters per CBS page (TS 123 041 [3]).
- The **broadcastArea** is an area (or a set of areas) that are the representation of the geographical area(s) to which a message is to be delivered. Areas can be defined by an alias name or a geometric shape such as circle or polygon.
  - In the case of an alias, the alias name shall be defined during provisioning and understood by both application and network.
  - In the case of a geometric shape, the network can restrict the maximum and minimum size of an area.

  Areas shall be covered by the mobile or wireline network. A message cannot be delivered to terminals that are out of network coverage even though the application has indicated the area(s) in the **sendBroadcastMessage** operation. If a specified area is out-of-coverage, the Message Broadcast Web Service will generate a **ServiceException** (SVC0300).

The application can also specify optional parameters as follows:

- **senderName** specifies the sender's name, i.e. the string that is displayed on the user's terminal as the originator of the message.

- **charging** specifies the charging information.
- **priority** specifies the priority of message.
- **deliveryTime** specifies the time at which message delivery should be initiated by the send operation. By using this parameter, the Web Service could avoid an overload condition through the scheduling of broadcasts at non-peak times.
- **totalBroadcasts** specifies how many times the message should be broadcasted to each of the designated area(s).
- **interval** specifies the time duration between the broadcasts. The minimum interval should be restricted by the network capabilities.
- **receiptRequest** is specified when the application requires to receive notification of the status of the broadcast message delivery. It is a **SimpleReference** structure that indicates the application endpoint, interface used for notification of delivery receipt and a correlator that uniquely identifies the sending request.
  - If the notification mechanism is not supported by a network, a **ServiceException** (SVC0283) will be returned to the application and the message will not be sent out to the area(s) specified.
  - 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.
  - Notification to the application is done by invoking the **notifyBroadcastDeliveryReceipt** operation at the endpoint specified in **receiptRequest**.
  - This optional message part is not used (or will be overridden) in case the **startDeliveryReceiptNotification** operation is used when the application requires to receive delivery receipt notifications. This is to avoid overlapping criteria.

The **result** part of the **sendBroadcastMessageResponse** message contains a request identifier for this message broadcast delivery request. The application can use it to invoke the **getBroadcastStatus** operation to poll for the delivery status.

8.1.1.1 Input message: sendBroadcastMessageRequest

Part name	Part type	Optional	Description
broadcastArea	BroadcastArea [1..unbounded]	No	Identifies the geographical area(s) to which a message is to be broadcasted
senderName	xsd:string	Yes	If present it indicates the message sender's 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 content of the message to be broadcasted
priority	MessagePriority	Yes	Priority of the message. If not present, the network will assign a priority based on an operator policy
deliveryTime	xsd:dateTime	Yes	If present, it specifies the time to initiate message broadcast in the network. If not present, the message is sent immediately
totalBroadcasts	xsd:int	Yes	The number of broadcasts. If not present, default value is 1.
interval	common:TimeMetric	Yes	The time difference between consecutive broadcasts. It is presents only if **totalBroadcasts** > 1
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 sent out to the designated area(s) or if delivery is impossible. It is not used (or will be overridden) in case startDeliveryReceiptNotification operation is used

8.1.1.2 Output message: sendBroadcastMessageResponse

Part name	Part type	Optional	Description
result	xsd:string	No	It identifies a specific message broadcast delivery request

8.1.1.3 Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.
- SVC0002: Invalid input value.
- SVC0280: Message too long.
- SVC0283: Delivery Status Notification not supported.
- SVC0300: Broadcast Area not supported.
- SVC0301: Too high load situation.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.
- POL0008: Charging not allowed.
- POL0330: Multiple areas not allowed.
- POL0331: Maximum Number of Areas exceeded.
- POL0332: Too many broadcasts requested.
- POL0333: Min /Max interval violation.

8.1.2 Operation: getBroadcastStatus

The application invokes the **getBroadcastStatus** operation to request the status information of a previous message broadcast request, i.e. **sendBroadcastMessage** operation, identified by **requestIdentifier**.

This operation 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 as defined by a service policy.

The operation returns a result set that can be expressed by a set of the broadcast status information, i.e. **BroadcastStatusInformation**, because multiple broadcast areas can be specified.

The result set may not include complete information, allowing the Web Service implementation to choose to deliver a partial set of results to accommodate other conditions, such as avoiding timeouts. In this case, the broadcast areas for which no attempt was made to provide data will be marked **NotRetrieved** in the result for each area.

BroadcastStatusInformation data structure, of course, includes the broadcast status(**BroadcastStatus**) of a specific area, supplemented with other data such as the number of broadcasts, success rate, broadcast end time. The presence of supplementary data varies according to the status value.

BroadcastStatus values have been identified as follows and the conceptual status diagram is shown in Figure 4.

- **MessageWaiting:** the message is still queued and not delivered to the network yet.
- **Broadcasting:** broadcasting is initiated and the network is still attempting to deliver the message as many times as requested in the **totalBroadcasts** of the **sendBroadcastMessageRequest** message. If **totalBroadcasts** = 1, this state will promptly transition to **Broadcasted** state after first delivery. If **totalBroadcasts** > 1, the state transition will not occur until the completion of delivery.

- **Broadcasted:** a final state indicating that the message was successfully delivered to the network as many times as requested.
- **BroadcastImpossible:** a final state indicating that delivering the broadcast message is impossible. The following reasons are possible:
  - Broadcast request was explicitly rejected by the network, because the designated area is out of network coverage.
  - Broadcast request was implicitly or explicitly rejected by the network, because the network is experiencing some problem such as overloads.
  - Broadcast request could not be delivered within the period specified by the **totalBroadcasts** and **interval** parts of the **sendBroadcastMessageRequest** message, and the message was discarded.
- **BroadcastUnknown:** indicating that the delivery state is unknown.
  - Network does not support a mechanism, temporarily or permanently, to find out the status even though the broadcast request has been delivered.
  - Delivery is pending: delivery is still being attempted because the period specified by the **totalBroadcasts** and **interval** parts of the **sendBroadcastMessageRequest** message has not expired yet.

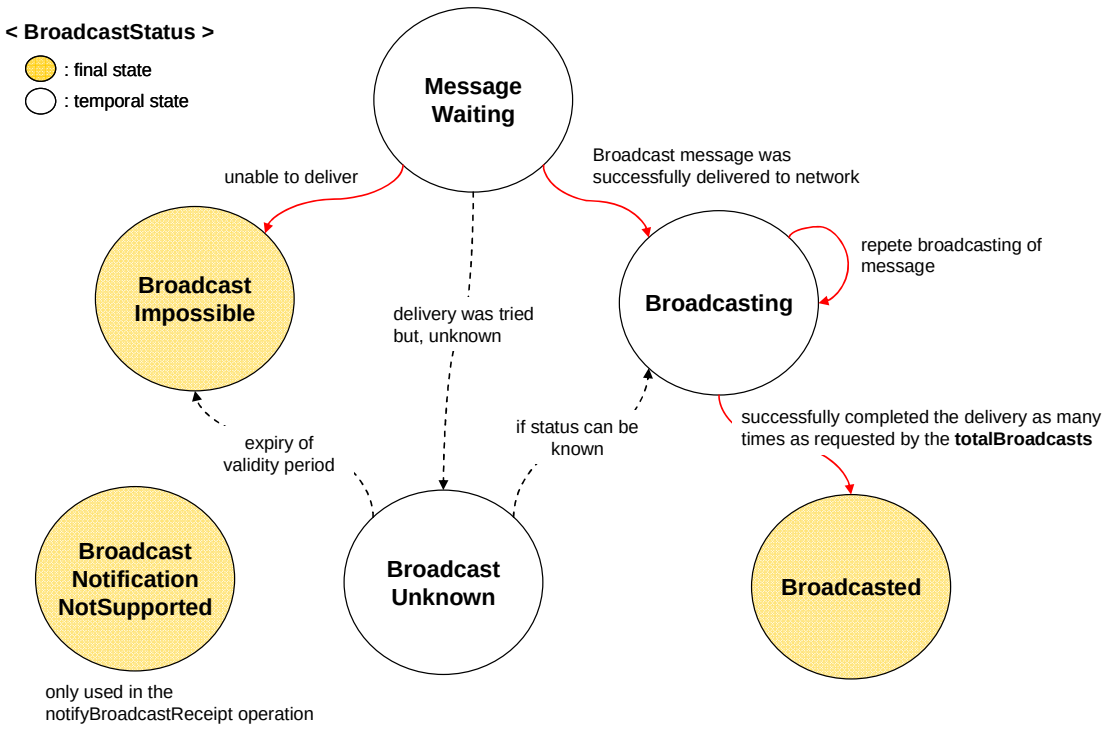


Figure 4: Conceptual status diagram for message broadcast

numberOfBroadcasts indicates the number of broadcasts, i.e. how many times the broadcast message has been successfully sent out to network. The value, of course, shall not exceed the value of the **totalBroadcasts** part of the **sendBroadcastMessageRequest** message. From this value, applications can figure out the current number of broadcast delivery which has been repeated so far.

successRate also provides another possible measurement of successful delivery and indicates what portion of the designated area has accepted the broadcast message request. In the case of mobile networks, this can be defined as the ratio of the number of BTSs that accepted the message and the total number of BTSs that should have accepted the message. Figure 5 shows an example. This is optional and present only if the broadcast status is either **Broadcasting** or **Broadcasted**. If the value is -1, it indicates the information is not available.

broadcastEndTime indicates the date and time at which the broadcast request completed. This value is optional and present only if the value of **BroadcastStatus** is **Broadcasted**.

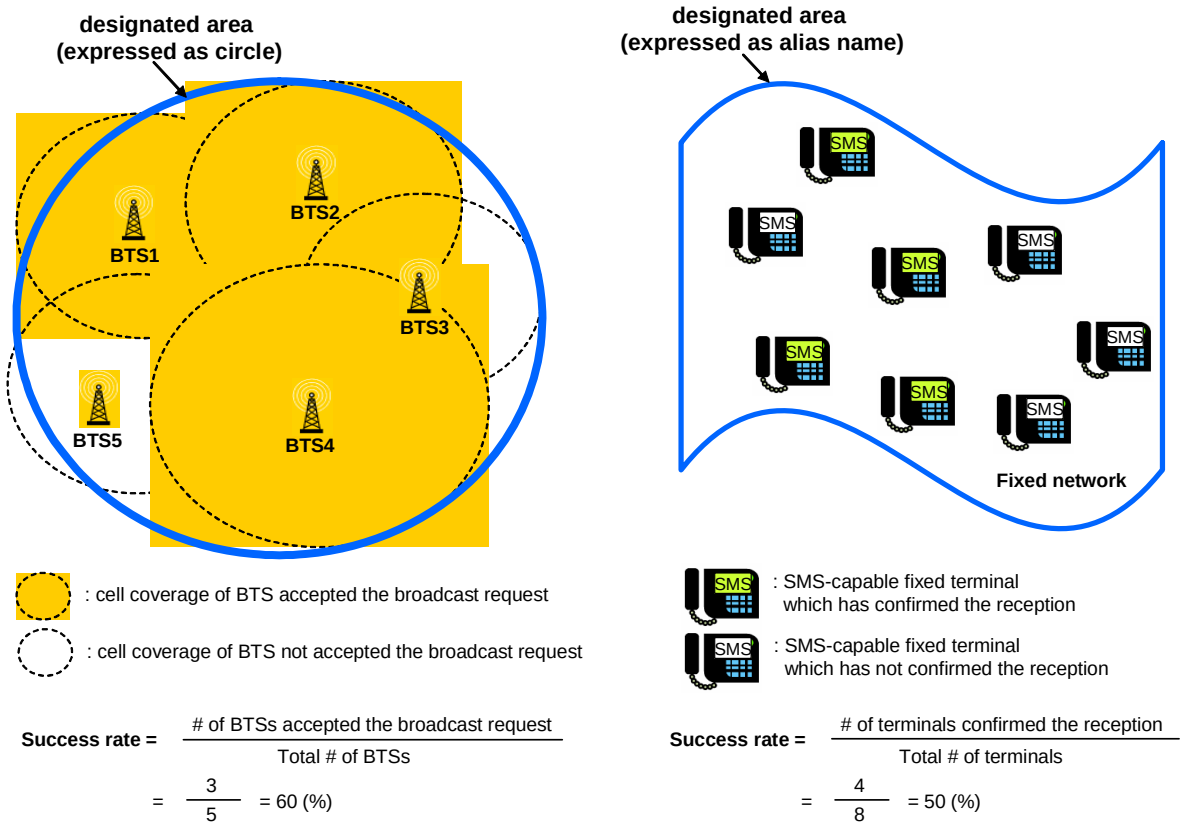


Figure 5: Success Rate

8.1.2.1 Input message: getBroadcastStatusRequest

Part name	Part type	Optional	Description
requestIdentifier	xsd:string	No	It identifies a specific **sendBroadcastMessage** request

8.1.2.2 Output message: getBroadcastStatusResponse

Part name	Part type	Optional	Description
result	BroadcastStatusData [1..unbounded]	No	Set of results for the request. It provides the broadcast status for each area with supplementary data such as the number of broadcasst, success rate, broadcast end time. Possible status values are: - MessageWaiting - Broadcasting - Broadcasted - BroadcastImpossible - BroadcastUnknown

8.1.2.3 Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.
- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.
- POL0010: Retention time interval expired.

8.1.3 Operation: cancelBroadcastMessage

The application invokes the **cancelBroadcastMessage** operation to request the cancellation of a previous **sendBroadcastMessage** request identified by **requestIdentifier**. It attempts to prevent the starting of a previous **sendBroadcastMessage** request and the restarting of **sendBroadcastMessage** request specified by the **totalBroadcasts** parameter. If this operation is invoked after initiating the N^{th} broadcast delivery, all the subsequent delivery after N^{th} delivery will be cancelled. In other words, it does not have any effect if the sending of broadcast message has already started.

8.1.3.1 Input message: cancelBroadcastMessageRequest

Part name	Part type	Optional	Description
requestIdentifier	xsd:string	No	It identifies a specific **sendBroadcastMessage** request.

8.1.3.2 Output message: cancelBroadcastMessageResponse

Part name	Part type	Optional	Description
None			

8.1.3.3 Referenced faults

ServiceException from ES 202 504-1 [2]:

- SVC0001: Service error.
- SVC0002: Invalid input value.

PolicyException from ES 202 504-1 [2]:

- POL0001: Policy error.

8.2 Interface: MessageBroadcastNotification

Message broadcast notification is the application side interface to which notifications about message broadcast status are delivered.

8.2.1 Operation: notifyBroadcastDeliveryReceipt

The **notifyBroadcastDeliveryReceipt** operation must be implemented by a Web Service at the ***application side*** if it requires notification of broadcast message delivery receipt. It will be invoked by the Parlay X server to notify the application when a message sent by an application has been handed over to a messaging center such as a CBC and subsequently sent out to the designated area(s), or if delivery is impossible. The notification will occur if and only if the status value of the sent message is **Broadcasted** or **BroadcastImpossible** and the application has specified interest in notification when sending a broadcast message using one of the following mutually exclusive mechanisms:

- By specifying the optional receiptRequest part. The correlator returned corresponds to the identifier specified by the application in the receiptRequest part.
- By invoking the **startDeliveryReceiptNotification** operation requesting to receive delivery receipt notifications. The correlator returned corresponds to the identifier specified by the application in the **reference** of the original **startDeliveryReceiptNofitication** request.

When a broadcast message is sent to multiple geographical areas, the server will send a notification for each area as and when a broadcast message is successfully sent out to that area.

The **notifyBroadcastDeliveryReceiptRequest** message includes a **statusInfo** part providing data about an area which includes report status, broadcast status, the number of broadcasts, success rate, broadcast end time. The **reportStatus** value shall be **Retrieved** and the possible **BroadcastStatus** values are as follows:

- **Broadcasted:** a final state indicating that the message was successfully delivered to the network as many times as requested.
- **BroadcastImpossible:** a final state indicating that delivering the broadcast message is impossible. The following reasons are possible:
  - Broadcast request was explicitly rejected by the network, because the designated area is out of network coverage.
  - Broadcast request was implicitly or explicitly rejected by the network, because the network is experiencing some problem such as overloads.
  - Broadcast request could not be delivered within the period specified by the **totalBroadcasts** and **interval** parts of the **sendBroadcastMessageRequest** message, and the message was discarded.
- **BroadcastNotificationNotSupported:** indicating that, while notification is supported by the network, it does not support broadcast receipt for this specific broadcast area.

The presence of the other data varies according to the above **BroadcastStatus** values. For more information refer to the description in clause 8.1.2.

8.2.1.1 Input message: notifyBroadcastDeliveryReceiptRequest

Part name	Part type	Optional	Description
correlator	xsd:string	No	It identifies the original request. This correlator was provided by the application in the **receiptRequest** part of the **sendBroadcastMessageRequest** message.
area	BroadcastArea	No	It indicates a geographical area specified in the original request.
statusInfo	BroadcastStatusInformation	No	It provides the broadcast status for the area, supplemented with data such as the number of broadcasts, success rate, broadcast end time. Possible status values are: - Broadcasted. - BroadcastImpossible. - BroadcastNotificationNotSupported.

8.2.1.2 Output message: notifyBroadcastDeliveryReceiptResponse

Part name	Part type	Optional	Description
None			

8.2.1.3 Referenced faults

None.

8.3 Interface: MessageBroadcastNotificationManager

The message broadcast notification manager enables applications to set up and tear down notifications for broadcast messages, online.

8.3.1 Operation: startDeliveryReceiptNotification

Start notifications to the application for delivery receipts. The reference will be where to sned the delivery receipts. The notifyBroadcastDeliveryReceipt method (see clause 8.2.1) must be implemented by a Web Service at the application side if it requires notification of broadcast message delivery receipt. When the **startDeliveryReceiptNotification** is supported by the Service Provider, its use overrides the delivery receipting mechanism supported in the SendBroadcastMessage API (see clause 8.1: sendBroadcastMessage operation).

8.3.1.1 Input message: startDeliveryReceiptNotificationRequest

Part name	Part type	Optional	Description
reference	common:SimpleReference	No	Notification endpoint definition
filterCriteria	xsd:string	No	The filterCriteria will allow the service to filter flexibly. One example would be for service provider to filter only successful final status. This however is implementation specific and will be left to the Service Provider.

8.3.1.2 Output message: startDeliveryReceiptNotificationResponse

Part name	Part type	Optional	Description
None			

8.3.1.3 Referenced faults

ServiceException from [6]:

- SVC0001 - Service error.
- SVC0002 - Invalid input value.
- SVC0005 - Duplicate correlator.
- SVC0008 - Overlapping Criteria.
- SVC0283 – Delivery Status Notification not supported.

PolicyException from [6]:

- POL0001 - Policy error.

8.3.2 Operation: stopDeliveryReceiptNotification

The application may end delivery receipt notification using this operation.

8.3.2.1 Input message: stopDeliveryReceiptNotificationRequest

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

8.3.2.2 Output message: stopDeliveryReceiptNotificationResponse

Part name	Part type	Optional	Description
None			

8.3.2.3 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

Refer to the definition in ES 202 504-4 [6].

9.1.2 SVC0283: Delivery Status Notification not supported

Refer to the definition in ES 202 504-4 [6].

9.1.3 SVC0300: Broadcast Area not supported

A specific area cannot be supported if, for example, the range of an area is out of network coverage.

Name	Description
messageId	SVC0300.
text	%1 area description cannot be supported by the network.
variables	%1 - Message part representing a broadcast area.

9.1.4 SVC0301: Too high load situation

Name	Description
messageId	SVC0301.
text	Network (e.g. messaging center) is in a overload situation. Retry after %1 minutes.
variables	%1 - suggested time duration by the network for next try.

9.2 PolicyException

9.2.1 POL0330: Multiple areas not allowed

Name	Description
messageId	POL0330
text	Multiple areas not allowed
variables	

9.2.2 POL0331: Maximum Number of Areas exceeded

Name	Description
messageId	POL0331
text	Maximum number of broadcast areas (%1) is exceeded
variables	%1 - maximum allowed number of broadcast areas

9.2.3 POL0332: Too Many Broadcasts requested

Name	Description
messageId	POL0332
text	Too many broadcasts requested. Maximum number of broadcasts allowed is %1
variables	%1 - allowed number of broadcasts

9.2.4 POL0333: Min/Max Interval Violation

Name	Description
messageId	POL0333
text	Minimum or Maximum interval violation. The possible min/max interval is between %1 seconds and %2 seconds
variables	%1 - Minimum possible interval value %2 – Maximum possible interval value

10 Service policies

Service policies for this service.

Name	Type	Description
ChargingSupported	xsd:boolean	Is charging supported for send operation.
MultipleAreaSupport	xsd:boolean	Multiple broadcast areas may be supported.
MaxNumberOfAreas	xsd:int	Maximum number of broadcast areas that can be requested in a send operation.
MinValueOfInterval	common:TimeMetric	Minimum value of interval in send operation.
MaxValueOfInterval	common:TimeMetric	Maximum value of interval in send operation.
MaxBroadcasts	xsd:int	Maximum number of broadcasts that can be requested in a send operation.
StatusRetentionTime	common:TimeMetric	A time interval that begins after the status of a broadcast message delivery request has reached a final value. During this interval, the delivery status information remains available for retrieval by the application.

Annex A (normative): WSDL for Message Broadcast

The document/literal WSDL representation of this interface specification is compliant to ES 202 504-1 [2] and is contained in text files (contained in archive es_20250415v010101m0.zip) which accompany the present document.

Annex B (informative): Bibliography

ETSI TR 121 905: "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Vocabulary for 3GPP Specifications (3GPP TR 21.905)".

History

Document history			
V1.1.1	February 2008	Membership Approval Procedure	MV 20080425: 2008-02-26 to 2008-04-25