Service Guide for Mobile Broadcast Services

Candidate Version 1.0 – 29 May 2007

Open Mobile Alliance

OMA-TS-BCAST_ServiceGuide-V1_0-20070529-C

Contents

Figures

Tables

1. Scope

The scope of this specification is the Service Guide functionality of OMA Mobile Broadcast Enabler. The Service Guide functionality is used to provide information about the services and contents available via broadcast channel, interaction channel or both.

The present document defines the Service Guide data model, the interfaces, the delivery and the discovery of the Service Guide functionality (including the declaration of Service Guides within a Service Guide) and the use of PreviewData for services switching, browsing and preview.

2. References

2.1 Normative References

[3GPP TS 22.022] "Personalisation of GSM Mobile Equipment (ME); Mobile functionality specification", 3rd Generation Partnership Project, Technical Specification 3GPP TS 22.022 Release 6,
URL: http://www.3gpp.orghttp:// www.3gpp.org/

[3GPP TS 23.003] "Numbering, Addressing and Identification", 3rd Generation Partnership Project TS 23.003 Release 6,
URL: http://www.3gpp.org/ftp/Specs/

[3GPP TS 26.346] "Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs", 3rd Generation Partnership Project TS 26.346 Release 6,
URL: http://www.3gpp.org/ftp/Specs/

[3GPP TS 33.220] "Generic Authentication Architecture, Generic Bootstrapping Architecture", 3rd Generation Partnership Project, Technical Specification 3GPP TS 33.220 Release 6,
URL: http://www.3gpp.orghttp:// www.3gpp.org/

[3GPP2 C.S0002-0] 3GPP2 C.S0002-0 "Physical Layer Standard for cdma2000 Spread Spectrum Systems", Release 0

[3GPP2 C.S0005-D] 3GPP2 C.S0024-D v2.0 "Upper Layer (Layer 3) Signaling Standard for cdma2000 Spread Spectrum Systems," Release D

[3GPP2 C.S0024-A] 3GPP2 C.S0024-A v2.0 "cdma2000 High Rate Packet Data Air Interface Specification", Release A

[3GPP2 C.S0054-0] 3GPP2 C.S0054-0 v2.0 "cdma000 High Rate Broadcast-Multicast Packet Data Air Interface Specification"

[BCAST10-BCMCS-Adaptation] "Broadcast Distribution System Adaptation – 3GPP2/BCMCS", Open Mobile Alliance™, OMA-TS-BCAST_BCMCS_Adaptation-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-Distribution] "File and Stream Distribution for Mobile Broadcast Services", Open Mobile Alliance™, OMA-TS-BCAST_Distribution-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-DVBH-IPDC-Adaptation] "Broadcast Distribution System Adaptation – IPDC over DVB-H", Open Mobile Alliance™, OMA-TS-BCAST_DVB_Adaptation-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-MBMS-Adaptation] "Broadcast Distribution System Adaptation – 3GPP/MBMS", Open Mobile Alliance™, OMA-TS-BCAST_MBMS_Adaptation-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-Requirements] "Mobile Broadcast Services Requirements", Open Mobile Alliance™, OMA-RD-BCAST-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-ServContProt] "Service and Content Protection for Mobile Broadcast Services", Open Mobile Alliance™, OMA-TS-BCAST_SvcCntProtection-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-Services] "Mobile Broadcast Services", Open Mobile Alliance™, OMA-TS-BCAST_Services-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-Schema-sg-f] "Mobile Broadcast Services – XML Schema for Service Guide Fragments", Open Mobile Alliance™, OMA-SUP-XSD_bcast_sg_fragments-V1_0,
URL: http://www.openmobilealliance.org/

[BCAST10-Schema- "Mobile Broadcast Services – XML Schema for Service Guide Backend Interfaces", Open

sg-be]	Mobile Alliance™, OMA-SUP-XSD_bcast_sg_backend-V1_0, URL: http://www.openmobilealliance.org/
[BCAST10-Schema-sg-sgdd]	"Mobile Broadcast Services – XML Schema for SGDD", Open Mobile Alliance™, OMA-SUP-XSD_bcast_sg_sgdd-V1_0, URL: http://www.openmobilealliance.org/
[DRM20-Broadcast-Extensions]	"OMA DRM v2.0 Extensions for Broadcast Support", Open Mobile Alliance™, OMA-TS-DRM-XBS-V1_0, URL: http://www.openmobilealliance.org/
[HTML401]	HTML 4.01 Specification, W3C Recommendation 24 December 1999, URL: http://www.w3.org/TR/html4/.
[IOPPROC]	"OMA Interoperability Policy and Process", Version 1.1, Open Mobile Alliance™, OMA-IOP-Process-V1_1, URL: URL:http://www.openmobilealliance.org/
[MIGFG]	The Moving Image Genre-Form Guide, Library of US Congress, URL: http:// http://www.loc.gov/rr/mopic/miggen.html
[OMA MLP]	"Mobile Location Protocol", Open Mobile Alliance™, OMA-TS-MLP-V3_2, URL:http://www.openmobilealliance.org/
[OMNA]	Open Mobile Naming Authority, Open Mobile Alliance(tm), URL: http://www.openmobilealliance.org/tech/omna
[RFC 1035]	"DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION", P. Mockapetris, November 1987, URL: http://www.ietf.org/rfc/rfc1035.txt
[RFC 1123]	"Requirements for Internet Hosts -- Application and Support", Internet Engineering Task Force, R. Braden, Editor, October 1989, URL: http://www.ietf.org/rfc/rfc1123.txt
[RFC 1305]	"RFC 1305 Network Time Protocol (Version 3) Specification, Implementation and Analysis", D. Mills, March 1992, URL: http://www.ietf.org/rfc/rfc1305.txt
[RFC 1952]	"GZIP file format specification version 4.3", P. Deutsch, May 1996, URL: http://www.ietf.org/rfc/rfc1952.txt
[RFC 2046]	"Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", N. Freed, N. Borenstein, November 1996, URL: http://www.ietf.org/rfc/rfc2046.txt
[RFC 2119]	"Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL: URL:http://www.ietf.org/rfc/rfc2119.txt
[RFC 2234]	"Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997, URL: URL:http://www.ietf.org/rfc/rfc2234.txt
[RFC 3066]	"Tags for the Identification of Languages", H. Alvestrand, January 2001, URL: http://www.ietf.org/http://www.ietf.org/rfc/rfc3066.txt
[RFC 3450]	"Asynchronous Layered Coding (ALC) Protocol Instantiation", M. Luby, J. Gemmell, L. Vicisano, L. Rizzo, J. Crowcroft, December 2002, URL: http://www.ietf.org/rfc/rfc3450.txt
[RFC 3451]	"Layered Coding Transport (LCT) Building Block", M. Luby, December 2002, URL: URL: http://www.ietf.org/rfc/rfc3451.txt
[RFC 3695]	"Compact Forward Error Correction(FEC) Schemes", Luby, M. and L. Vicisano, URL: URL: http://www.ietf.org/rfc/rfc3695.txt
[RFC 3926]	"FLUTE - File Delivery over Unidirectional Transport", T. Paila, October 2004,

URL: URL: http://www.ietf.org/rfc/rfc3926.txt

[RFC 4566] "SDP – Session Description Protocol", M. Handley, July 2006,
URL: URL: http://www.ietf.org/rfc/rfc4566.txt

[RFC2246] "The TLS Protocol, Version 1.0", T. Dierks, C.Allen, January 1999,
URL: http://www.ietf.org/rfc/rfc2246.txt

[SSL30] "SSL 3.0 Specification", Netscape Communications, November 1996,
URL: http://wp.netscape.com/eng/ssl3/draft302.txt

[TVA-Metadata] ETSI TS 102 822-3-1 v1.3.1, "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime")",
URL: http://portal.etsi.org

[XML] Extensible Markup Language (XML) 1.1, W3C Recommendation 04 February 2004, edited in place 15 April 2004
URL: . http://www.w3.org/TR/xml11

[XML-Schema-2] XML Schema Part 2: Datatypes, Second Edition, W3C Recommendation 28 October 2004,
URL: http://www.w3.org/TR/xmlschema-2/#built-in-datatypes

2.2 Informative References

[BCAST10-Architecture] "Mobile Broadcast Services Architecture", Open Mobile Alliance™, OMA-AD- BCAST-V1_0,
URL: http://www.openmobilealliance.org/

[RFC 2048] "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", N. Freed, J. Klensin, and J. Postel, November 1996,
URL: http://www.ietf.org/rfc/rfc2048.txt

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

The following is the legend used in this specification:

Type: E=Element, A=Attribute, E1=sub-element, E2=sub-element’s sub-element, E[n]=sub-element of element[n-1]

Category: NM = Mandatory for network to support; NO = Optional for network to support; TM = Mandatory for terminal to support; TO = Optional for terminal to support

Cardinality: x..y = the number of the presented instance of this element/attribute is in the range from x to y. If x=0, this specific element/attribute is OPTIONAL for network to use, otherwise it is MANDATORY for network to use.

3.2 Definitions

Announcement session	Transport sessions which represent the Service Guide Announcement Channel.
Auxiliary Data	Various types of media content, such as text, static image, audio or video clips, etc. used to customize a service or content item, or to provide enhancements or supplementary information to that service or content.
Barker	A content item containing media that is presented to the user in place of a selected content item to which the user is not subscribed.
BSMCode	An identifier stored on the BCAST terminal, representing the BSM of the Broadcast Service Provider. For example if the BSMCode is included in the ‘PurchaseChannel’ fragment, it informs the terminal whether the purchase items belonging to that purchase channel may be purchased by the user. BSMCode is presently defined as corresponding to either Smartcard-based terminals or non-Smartcard based terminals.
Cachecast	A non real-time file distribution service, for which the content could consist of audio, audio and video, and/or other types of data. Once the subscriber has subscribed to this service, the content is delivered to the subscriber’s mobile device in the background, during the Distribution Window, transparently to the user. The media is stored on the device and may be accessed by the user during a scheduled availability period referred to as the Presentation Window.
Entry Point	Connectivity related information required by the BCAST Terminal to access service/content, Service Guide, or Service Guide Delivery Descriptor.
GZIP	Abbreviation for GNU zip, a file compression algorithm defined by IETF RFC 1952, “GZIP file format specification version 4.3”.
Preview Data	Various types of media content, such as text, static image, audio clips, video clips, etc, or the combination of the above mentioned media components, used to present the outline of a service, content or purchase item to users. The preview data is described by the ‘PreviewData’ fragment of the Service Guide.
Service Class	A unique name used to identify a particular broadcast service. The service class is a combination of many properties, such as media formats or file types under use, handling of media, interaction with user, etc., used collectively to describe how a particular service works.
Service Guide Announcement Channel	A broadcast distribution channel over which Service Guide Delivery Descriptors carried within announcement sessions, are delivered to the terminal.
Service Guide Delivery Channel	A broadcast distribution channel over which Service Guide Delivery Units are delivered to the terminal.
Service Guide Fragment	An atomic information component of the Service Guide, which can be compressed, encapsulated and transported in the absence of other parts of the Service Guide.

3.3 Abbreviations

3GPP	3rd Generation Partnership Project
ALC	Asynchronous Layered Coding
BCAST	Mobile Broadcast Services
BCMCS	Broadcast Multicast Service
BDS	Broadcast Distribution System
BSA	BCAST Service Application
BSD/A	BCAST Service Distribution/Adaptation
BSM	BCAST Subscription Management
CID	Content ID
DRM	Digital Rights Management
DVB	Digital Video Broadcast
DVB-H	Digital Video Broadcast – Handheld
EXT_CENC	(LCT header) Extension (defining) Content Encoding
EXT_FTI	(LCT header) Extension (defining) FEC Transmission Information
FDT	File Delivery Table
FEC	Forward Error Correction
FLUTE	File Delivery over Unidirectional Transport
GBA	Generic Bootstrapping Architecture
GBA_ME	Mobile Equipment (Terminal) based GBA
GBA_U	GBA with UICC-based enhancements
GZIP	GNU zip
HRPD	High Rate Packet Data
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
IPDC	IP DataCast
KMS	Key Management System
LCT	Layered Coding Transport
MBMS	Multimedia Broadcast / Multicast Service
MD5	Message Digest (Number) 5
MIME	Multipurpose Internet Mail Extensions
MLP	Mobile Location Protocol
MMS	Multimedia Messaging Service
NID	Network Identification
NTP	Network Time Protocol
OMA	Open Mobile Alliance
OTI	Object Transmission Information
PZID	Packet Zone ID
RI	Rights Issuer

RO	Rights Object
RTP	Real-time Transport Protocol
RTSP	Real-Time Streaming Protocol
SDP	Session Description Protocol
SG	Service Guide
SG-A	Service Guide Adaptation
SG-C	Service Guide-Client
SG-D	Service Guide-Distribution
SGDD	Service Guide Delivery Descriptor
SGDU	Service Guide Delivery Unit
SG-G	Service Guide Generation
SID	System Identification
SMIL	Synchronized Multimedia Integration Language
SMS	Short Message Service
TOI	Transport Object Identifier
TSI	Transport Session Identifier
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USBD	User Service Bundle Description
USD	User Service Description
WAP	Wireless Access Protocol
XML	Extensible Markup Language

4. Introduction

This document specifies the Service Guide Function. Service Guide enables the service and content providers to describe the services and content they make available, or offer for subscription or purchase, as Mobile Broadcast services either over Broadcast Channel or over Interaction Channel. It also enables the way to describe how to access the services. From the user perspective the Service Guide can be seen as an entry point to discover the currently available or scheduled services and content and to filter those based on their preferences. Furthermore, the Service Guide provides the entry point to interactive services.

The Service Guide comprises of data model that models the services, schedules, content, related purchase and provisioning data, access and interactivity data in terms of Service Guide fragments. Further, this specification defines methods for initial discovery of Service Guide and the declaration of Service Guide fragments through Service Guide Delivery Descriptors that are sent over Service Guide Announcement Channel. The delivery methods are specified for Service Guide fragment delivery over Service Guide Delivery Channel, both over the Broadcast Channel as well as over the Interactive Channel. To manage the Service Guide data, the specification defines update and management methods. Finally, this document specifies the relevant back-end interfaces for Service Guide.

5. Service Guide

5.1 Service Guide Data Model

5.1.1 Service Guide Structure and Fragmentation

The structure of the Service Guide data model is illustrated in the Figure 1 below. The basic assumption is that the data is represented as XML fragments. Each XML fragment is considered as a separate well-formed XML document [XML]. The XML text declaration MAY be omitted. In such a case, the terminal SHALL assume the following default XML text declaration to ensure well-formedness:

```
<?xml version="1.1"?>
```

The namespaces used in a fragment SHOULD be declared in the fragment according to XML rules [XML]. If no namespace is declared, the terminal SHALL assume that the default namespace of the fragment is “urn:oma:xml:bcast:sg:fragments:1.0”.

The meaning of the cardinalities shown in the Figure 1 is the following: One instantiation of Fragment A as in Figure 2 references c to d instantiations of Fragment B. If c=d, d is omitted. Thus, if c > 0 and Fragment A exists, at least c instantiation of Fragment B must also exist, but at most d instantiations of Fragment B may exist. Vice versa, one instantiation of Fragment B is referenced by a to b instantiations of Fragment A. If a=b, b is omitted. The arrow connection from Fragment A pointing to Fragment B indicates that Fragment A contains the reference to Fragment B.

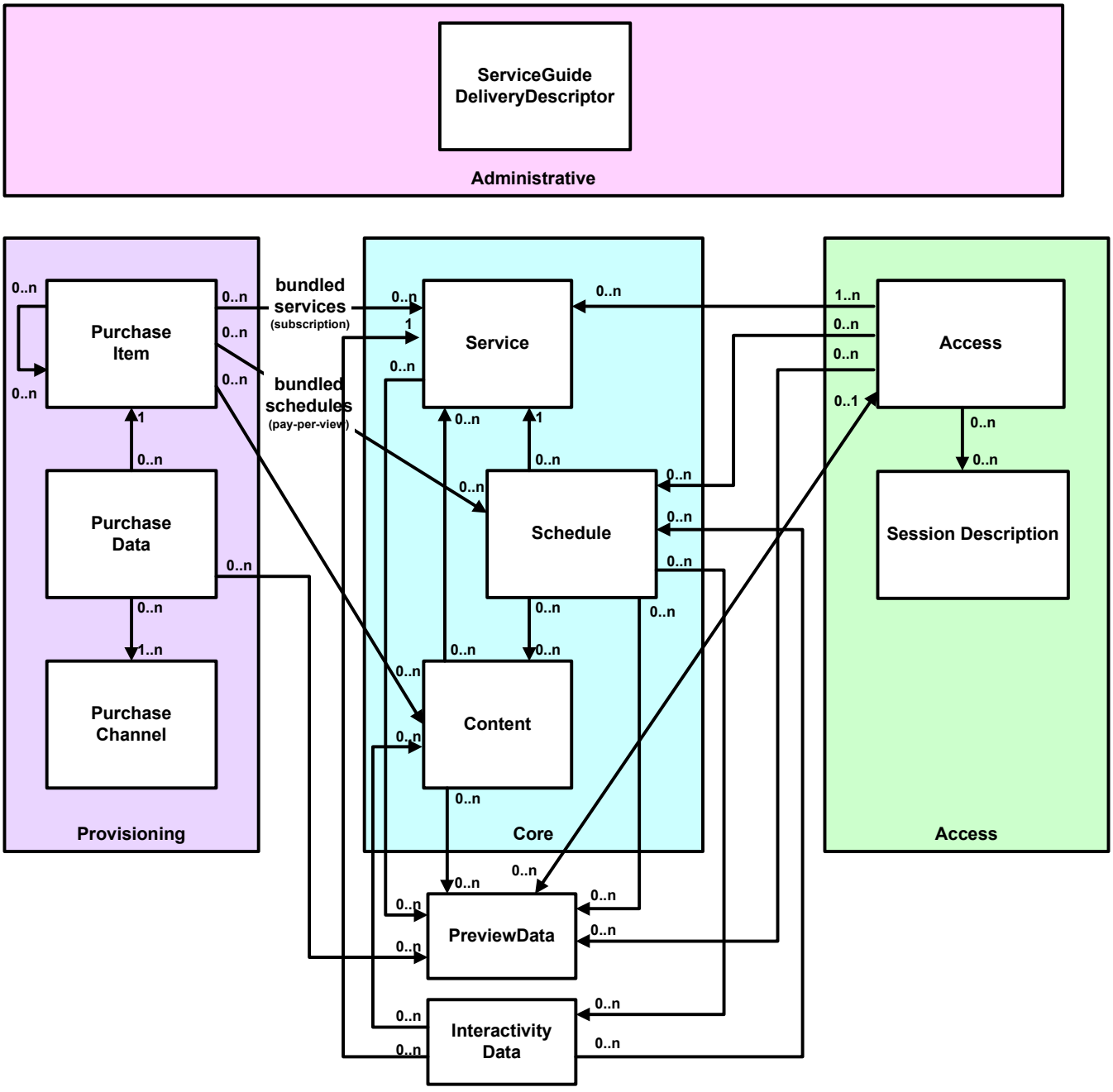


Figure 1: Structure of Service Guide

Any given 'PurchaseItem' fragment SHALL only be able to reference a single type among the following fragments: 'Service', 'Schedule', 'Content', or another 'PurchaseItem'. 'Access' fragment SHALL have a link to either 'Service' fragment or 'Schedule' fragment.

As shown in Figure 1, all of the connection arrows between Service Guide fragments are uni-directional, specially, there are two pairs of opposite uni-directional arrows: one pair is between 'Schedule' fragment and 'InteractivityData' fragment, and the other pair is between 'Acces' fragment and 'PreviewData' fragment. The reference arrow from 'Schedule' fragment to 'InteractivityData' fragment declares the distribution schedule of the interactive media documents carried in a file stream (referenced by the 'InteractivityData' fragment); the reference arrow from 'InteractivityData' fragment to 'Schedule' fragment declares which 'Schedule' Fragment this 'InteractivityData' Fragment is associated with. The reference arrow from 'Access' fragment to 'PreviewData' fragment indicates the service-by-serivce switching preview information for the access; the reference arrow from 'PreviewData' fragment to 'Access' fragment declares how the preview data can be accessed.

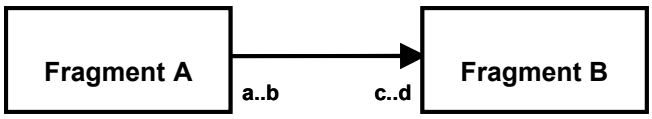


Figure 2: Cardinalities and Reference Direction between Service Guide Fragments

The semantics of the elements in the model as defined as follows:

Service

The 'Service' fragment describes at an aggregate level the content items which comprise a broadcast service.. The service may be delivered to the user using multiple means of access, for example, the broadcast channel and the interactive channel. The service may be targeted at a certain user group or geographical area. Depending on the type of the service it may have interactive part(s), broadcast-only part(s), or both.

Further, the service may include components not directly related to the content but to the functionality of the service – such as purchasing or subscription information. As the part of the Service Guide, the 'Service' fragment forms a central hub referenced by the other fragments including 'Access', 'Schedule', 'Content' and 'PurchaseItem' fragments. In addition to that, the 'Service' fragment may reference 'PreviewData' fragment. It may be referenced by none or several of each of these fragments.

Together with the associated fragments the terminal may determine the details associated with the service at any point of time. These details may be summarized into a user-friendly display, for example, of what, how and when the associated content may be consumed and at what cost

Schedule

The 'Schedule' fragment defines the timeframes in which associated content items are available for streaming, downloading and/or rendering. This fragment always references the 'Service' fragment. If it also references one or more 'Content' fragments or 'InterativityData' fragments, then it defines the valid distribution and/or presentation timeframe of those content items belonging to the service, or the valid distribution timeframe and the automatic activation time of the InteractivityMediaDocuments associated with the service. On the other hand, if the 'Schedule' fragment does not reference any 'Content' fragment(s) or 'InteractivityDat'a fragment(s), then it defines the timeframe of the service availability which is unbounded.

Content

The 'Content' fragment gives a detailed description of a specific content item. In addition to defining a type, description and language of the content, it may provide information about the targeted user group or geographical area, as well as genre and parental rating.

The 'Content' fragment may be referenced by Schedule, PurchaseItem or 'InteractivityData' fragment. It may reference 'PreviewData' fragment or 'Service' fragment.

Access

The 'Access' fragment describes how the service may be accessed during the lifespan of the service. This fragment contains or references Session Description information and indicates the delivery method. One or more 'Access' fragments may reference a 'Service' fragment, offering alternative ways for accessing or interacting with the associated service.

For the Terminal, the 'Access' fragment provides information on what capabilities are required from the terminal to receive and render the service. The 'Access' fragment provides Session Description parameters either in the form of inline text, or through a pointer in the form of a URI to a separate Session Description. Session Description information may be delivered over either the broadcast channel or the interaction channel.

SessionDescription

The 'SessionDescription' is a Service Guide fragment which provides the session information for access to a service or content item. Further, the Session Description may provide auxiliary description information, used for associated delivery procedures.

The Session Description information is provided using either syntax of SDP in text format, or through a 3GPP MBMS User Service Bundle Description [3GPP TS 26.346] (USBD).

Auxiliary description information is provided in XML format and contains an Associated Delivery Description as specified in [BCAST10-Distribution].

Note that in case SDP syntax is used, an alternative way to deliver the Session Description is by encapsulating the SDP in text format in 'Access' fragment.

Note that Session Description as a concept may be used both for Service Guide delivery itself as well as for the content sessions.

PurchaseItem

The 'PurchaseItem' fragment represents a group of one or more services (i.e. a service bundle) or one or more content items, offered to the end user for free, for subscription and/or purchase.

This fragment can be referenced by 'PurchaseData' fragment(s) offering more information on different service bundles. The 'PurchaseItem' fragment may be also associated with:

- a 'Service' fragment to enable bundled services subscription and/or,
- a 'Schedule' fragment to enable consuming a certain service or content in a certain timeframe (pay-per-view functionality) and/or,
- a 'Content' fragment to enable purchasing a single content file related to a service.
- other 'PurchaseItem' fragments to enable bundling of purchase items

PurchaseData

The main function of the 'PurchaseData' fragment is to express all the available pricing information about the associated purchase item.

The 'PurchaseData' fragment collects the information about one or several purchase channels and may be associated with PreviewData specific to a certain service or service bundle. It carries information about pricing of a service, a service bundle, or, a content item. Also, information about promotional activities may be included in this fragment.

PurchaseChannel

The 'PurchaseChannel' fragment carries the information about the entity from which purchase of access and/or content rights for a certain service, service bundle or content item may be obtained, as defined in the 'PurchaseData' fragment. The purchase channel is associated with one or more Broadcast Subscription Managements (BSMs). The terminal is only permitted to access a particular purchase channel if it is affiliated with a BSM that is also associated with that purchase channel.

Multiple purchase channels may be associated to one 'PurchaseData' fragment. A certain end-user can have a "preferred" purchase channel (e.g. his/her mobile operator) to which all purchase requests should be directed. The preferred purchase channel may even be the only channel that an end-user is allowed to use.

PreviewData

'PreviewData' fragment contains information that is used by the terminal to present the service or content outline to users, so that the users can have a general idea of what the service or content is about. 'PreviewData' fragment can include simple texts, static images (for example, logo), short video clips, or even reference to another service which could be a low bit rate version for the main service. 'Service', 'Content', 'PurchaseData', 'Access' and 'Schedule' fragments may reference 'PreviewData' fragment

InteractivityData

The InteractivityData contains information that is used by the terminal to offer interactive services to the user, which is associated with the broadcast content. These interactive services enable users to e.g. vote during TV shows or to obtain content related to the broadcast content. 'InteractivityData' fragment points to one or many 'InteractivityMedia' documents that include xhtml files, static images, email template, SMS template, MMS template documents, etc. The 'InteractivityData' fragment may reference the 'Service', 'Content' and 'Schedule' fragments, and may be referenced by the 'Schedule' fragment.

ServiceGuideDeliveryDescriptor

The ServiceGuideDeliveryDescriptor is transported on the Service Guide Announcement Channel, and informs the terminal the availability, metadata and grouping of the fragments of the Service Guide in the Service Guide discovery process (see section 6.1.1). A SGDD allows quick identification of the Service Guide fragments that are either cached in the terminal or being transmitted. For that reason, the SGDD is preferably repeated if distributed over broadcast channel. The SGDD also provides the grouping of related Service Guide fragments and thus a means to determine completeness of such group.

The ServiceGuideDeliveryDescriptor is especially useful if the terminal moves from one service coverage area to another. In this case, the ServiceGuideDeliveryDescriptor can be used to quickly check which of the Service Guide fragments that have been received in the previous service coverage area are still valid in the current service coverage area, and therefore don't have to be re-parsed and re-processed.

5.1.2 Service Guide Elements, Attributes and Meta Data

This section defines the Service Guide metadata, including elements and attributes of every Service Guide fragment and the type, cardinality, category, description and data type of each element and attribute.

In 'Service' fragment, 'Content' fragment, 'PurchaseItem' fragment, 'PurchaseData' fragment and 'PurchaseChannel' fragment, there are some elements or attributes which are particularly intended for end-user as program guide. This kind of metadata are grouped and highlighted between the "Start of program guide" and "End of program guide" cells in the above mentioned fragments. The program guide elements or attributes SHOULD be used for presentation purposes, and MAY be used for terminal filtering, search, sort, etc. Other elements or attributes can be used for presentation only, for terminal interpretation only, or for both.

Every SG fragments specified in this section may have zero, one or more than one instances in a Service Guide.

The tables specifying each Service Guide fragment in the following sections use various data types. The data types SHALL conform to [XML-Schema-2].

Terminals with interaction channel (e.g. UMTS, GPRS) SHALL support all SG fragments. Terminals with no return channel MAY NOT support the 'InteractivityData' fragment, but SHALL support all the other SG fragments.

Network SHOULD support and MAY use all SG fragments.

The Service Guide Data Model is formally reflected as XML Schema in [BCAST10-Schema-sg-f].

The terminal SHALL NOT offer an interface that allows modification or removal of any information contained in the service guide. As an exception it MAY be possible to remove whole fragments and the information contained therein, however then it SHALL NOT be possible to access or render the service or program described therein anymore. It MAY be possible to add fragments to the service guide. [*]

[*] The normative statements in this paragraph are not intended to be a part of the SCR tables.

5.1.2.1 Service

A service represents a bundle of content items, which forms a logical group to the end-user. An example would be a TV channel, composed of several TV shows. A 'Service' fragment contains the metadata describing the Mobile Broadcast service. It is possible that the same metadata (i.e., attributes and elements) exist in the 'Content' fragment(s) associated with that 'Service' fragment. In that situation, for the following elements: 'ParentalRating', 'TargetUserProfile', 'Genre' and 'BroadcastArea', the values defined in 'Content' fragment take precedence over those in 'Service' fragment.

Name	Type	Category	Cardinality	Description	Data Type
Service	E			'Service' fragment Contains the following attributes: id version validFrom validTo weight serviceContentProtection baseCID emergency Contains the following elements: ProtectionKeyID ServiceType GlobalServiceID Name Description AudioLanguage TextLanguage ParentalRating TargetUserProfile Genre Extension PreviewDataReference BroadcastArea TermsOfUse PrivateExt	
id	A	NM/ TM	1	ID of the 'Service' fragment. The the value of this attribute SHALL be globally unique."	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no 'validFrom' attribute is given. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future.	unsignedInt

weight	A	NM/ TM	0..1	Intended order of display of this service relative to other services as presented to the end user. The order of display is by increasing weight value (i.e., service with lowest weight is displayed first). Default: 65535 User preference, if available, SHALL override the weight.	unsignedShort
serviceContentProtection	A	NO/ TO	0..1	Specifies if the service is encrypted (true) or not (false). This element will be used for presentation purpose to users. Default: false	boolean
baseCID	A	NO/ TO	0..1	For the DRM Profile, part of the Service or Program CID used to identify the corresponding asset within a OMA DRM 2.0 Rights Object. The Service or Program CID is obtained from the BaseCID as described in [BCAST10-ServContProt] section 5.5.1. This element is only Mandatory to support for the network and terminal in case the DRM Profile is supported [BCAST10-ServContProt]. Note: for uniqueness of the baseCID see Appendix H.	string
emergency	A	NO/ TO	0..1	When assigned with value 'true', specifies that this service is a service of emergency nature. That also means that all content items belonging to this service are contents of emergency nature. This attribute can be used for presentation purposes to users. It is RECOMMENDED that the Terminal processes the reception of the services or content of emergency nature with high priority, and highlights their availability to user. How to order the emergency service or content is out of the scope of the specification. The default value of this attribute is 'false'.	boolean
ProtectionKeyID	E1	NO/ TO	0..N	List of key identifiers needed to access protected content. This information allows the terminal to determine whether or not it has the correct key material to access services within a PurchaseItem. In a scenario where this fragment is shared among multiple service providers, different key identifiers from the different service providers to access this specific protected service/content may be mixed in this element and the terminal SHOULD be able to sort out the key identifiers associated with the terminal's affiliated service provider based on the Key Domain ID. How this is used is out of scope and is left to implementation. The network and terminal SHALL support this element in case the Smartcard Profile is supported [BCAST10-ServContProt].	base64Binary

				The protection key identifiers to access a specific service or content item SHALL only be signalled in one of the following fragment types: 'Service', 'Content', 'PurchaseItem' or 'Access' fragments, but not mixed. Contains the following attribute: type	
type	A	NM/TM	1	Type of ProtectionKeyID: 0: ProtectionKeyID = Key Domain ID concatenated with SEK/PEK ID, where both values are as used in the Smartcard Profile [BCAST10-ServContProt]. 1-127 Reserved for future use 128-255 Reserved for proprietary use	unsignedByte
ServiceType	E1	NM/ TM	0..N	Type of the service. Allowed values are: 0 - unspecified 1 - Basic TV 2 - Basic Radio 3 - RI services 4 - Cachecast 5 - File download services 6 - Software management services 7 - Notification 8 – Service Guide 9 - Terminal Provisioning services 10 - 127 reserved for future use 128 -255 reserved for proprietary use The mixed service types SHALL be indicated by the presence of multiple instances of ServiceType (for example, for mixed Basic TV and Cachecast, two instances of ServiceType, with values 1 and 4 are present for this 'Service' fragment. This element SHALL be processed by the terminal strictly for rendering to the user for example as a textual indicator, an icon, or graphic representation for the service. However, 'ServiceType' with value of 3 and 9 SHALL NOT be rendered and their existence SHOULD NOT be displayed to the user. With value 6, i.e. sofware management services, users can select the desired software components (eg. desktop theme, ringtone, SG navigator update) to download over broadcast channel or interaction channel. The software components provided by this sofware management service are described by 'Content' fragments which belong to this 'Service' fragment. It is not expected that terminals are able to automatically select and download software components using this type of service.	unsigned Byte

GlobalServiceID	E1	NM/ TM	0..1	The globally unique identifier identifying the service this 'Service' fragment describes.	anyURI
				Start of program guide The program guide elements of this fragment are grouped between the Start of program guide and end of program guide cells in this fragment. The program guide elements are for user interpretation. This enables the content creator to provide user readable information about the service. The terminal SHOULD use all declared program guide elements in this fragment for presentation to the end-user. The terminal MAY offer search, sort etc functionalities. The Program Guide consists of the following elements: Name Description AudioLanguage TextLanguage ParentalRating TargetUserProfile Genre Extension	
Name	E1	NM/ TM	1..N	Name of the Service, possibly in multiple languages.The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
Description	E1	NM/ TM	0..N	Description, possibly in multiple languages.The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
AudioLanguage	E1	NM/ TM	0..N	This element declares for the end users that this service is available with an audio track corresponding to the language represented by the value of this element. The textual value of this element can be made available for the end users in different languages. In such a case the language used to represent the value of this element is signalled using the built-in XML attribute 'xml:lang'. See section 7, Multi-language support. As the instances of this element are representing choices w.r.t. language for the end users, the network needs to make sure the choices are unambiguous. Therefore there are certain restrictions on the values of the attributes 'id' and 'xml:lang'. The former are defined below and for the latter they are as follows: If the network instantiates the 'AudioLanguage' the following applies: • The value of the 'id' attribute SHALL be the same for 'Audiolanguage' elements that declare the same audio	string

				stream, but are made available to the end-user in different textual languages (different 'xml:lang' attributes). • For the case above, only one 'Audiolanguage' element can have the 'xml:lang' attribute omitted. • A particular 'id' value SHALL not be used to declare more than one audio streams for this fragment. If the attribute 'id' is omitted then terminal SHALL interpret this language setting being applicable for all the consumption methods available for the 'Service' in question. The network SHALL instantiate this element multiple times if and only if each of the instances contains the 'id' attribute. Contains the following attributes: id languageSDPTag	
id	A	NM/ TM	0..1	Identifier of the element 'AudioLanguage'. This is used to identify and select specific audio language upon reference by a 'Schedule' fragment. See section 7 about 'Multi-language support' and attribute 'audioLanguageIDRef' in 'Schedule' fragment. In case the network instantiates a 'Service' fragment with more than one instance of 'AudioLanguage' elements, the following constraints SHALL apply for any two of the instances: If the instances represent different choices of languages, the instances MUST be assigned different values of this attribute, but if the instances represent the same choice of language, the instances MUST be assigned the same value for this attribute.	anyURI
languageSDPTag	A	NM/ TO	1	Identifier of the audio language described by the parent 'AudioLanguage' element as used in the media sections describing the audio track in a Session Description. • The 'languageSDPTag' SHALL be formatted according to the rules of [RFC 3066], for the described language. • Each 'AudioLanguage' element declaring the same audio stream SHALL have the same value of the 'languageSDPTag'.	string
TextLanguage	E1	NM/ TM	0..N	This element declares for the end user that the textual components of this service are available in the language represented by the value of this element. The textual components can be, for instance, a caption or a sub-title track.	string

				The textual value of this element can be made available for the end users in different languages. In such a case the language used to represent the value of this element is signalled using the built-in XML attribute 'xml:lang'. See section 7 Multi-language support. The same rules and constraints as specified for the element 'AudioLanguage' of assigning and interpreting the attributes 'id', 'languageSDPTag' and 'xml:lang' SHALL be applied for this element also. If the attribute 'id' is omitted then terminal SHALL interpret this language setting being applicable for all the consumption methods available for the 'Service' in question. The network SHALL instantiate this element multiple times if and only if each of the instances contains the 'id' attribute. Contains the following attributes: id languageSDPTag	
id	A	NM/ TM	0..1	Identifier of the element 'TextLanguage'. This is used to identify and select specific language upon reference by a 'Schedule' fragment. See section 7 about 'Multi-language support' and attribute 'textLanguageIDRef' in 'Schedule' fragment.	anyURI
languageSDPTag	A	NM/TO	1	Identifier of the text language described by the parent 'TextLanguage' element as used in the media sections describing the textual track in a Session Description.	string
ParentalRating	E1	NM/ TM	0..N	The rating level defining criteria parents might use to determine whether the associated item is suitable for access by children, defined according to the regulatory requirements of the service area. Contains the following attribute: ratingSystem	string
ratingSystem	A	NO/ TO	0..1	Specifies the parental rating system in use, in which context the value of 'ParentalRating' element is semantically defined. Absence of this attribute means that the rating system is undefined.	string
TargetUserProfile	E1	NO/ TO	0..N	Profile attributes of the users whom the service is targeting at. The detailed personal attribute names and the corresponding values are specified by attributes of 'attributeName' and 'attributeValue'. Amongst the possible profile attribute names are age, gender, occupation, etc. (subject to national/local rules & regulations, if present and as applicable regarding use of personal profiling information and personal data privacy). The extensible list of 'attributeName' and 'attributeValue' pairs for a particular service enables end user profile filtering and end user preference filtering of broadcast services. The	

				terminal SHOULD be able to support 'TargetUserProfile' element. The terminal behavior for interpreting and acting upon 'TargetUserProfile' is out of the scope. It is RECOMMENDED that use of 'TargetUserProfile' element is an "opt-in" capability for users. Terminal settings SHOULD allow users to configure whether to input their personal profile or preference and whether to allow broadcast service to be automatically filtered based on the users' personal attributes without users' request. Contains the following attributes: attributeName attributeValue	
attributeName	A	NO/ TM	1	Profile attribute name	string
attributeValue	A	NO/ TM	1	Profile attribute value	string
Genre	E1	NM/ TM	0..N	Classification of service associated with characteristic form (e.g. comedy, drama). The OMA BCAST Service Guide allows describing the format of the Genre element in the Service Guide in two ways: • The first way is to use a free string • The second way is to use the "href" attributes of the Genre element to convey the information in the form of a controlled vocabulary (classification scheme as defined in [TVA-Metadata] or classification list as defined in [MIGFG]). The built-in XML attribute xml:lang MAY be used with this element to express the language. The Network MAY instantiate several different sets of 'Genre' element, using it as a free string or with a 'href' attribute. The Network SHALL ensure the different sets have equivalent and non-conflicting meaning, and the terminal SHALL select one of the sets to interpret for the end-user. Contains the following attributes: type href	string
type	A	NO/ TO	0..1	This attribute signals the level of this 'Genre' element. The following values are allowed: "main" "secondary" "other"	string
href	A	NO/	0..1	This attribute signals the controlled vocabulary	anyURI

		TO		used for this ‘Genre’ element. If this attribute is supported, terminals MAY support the following classification schemes according to [TVA-Metadata]: • for values of the ‘type’ attribute equal to "main" or "secondary", the terminal MAY support levels 1-4 of the TV Anytime ContentCS classification scheme identified by the classificationSchemeURI urn:tva:metadata:cs:ContentCS:2005 as defined in Annex A.8 of [TVA-Metadata] • for a value of the ‘type’ attribute equal to "other", the terminal MAY support levels 1-3 of the TV Anytime IntendedAudienceCS classification scheme identified by the classificationSchemeURI urn:tva:metadata:cs:IntendedAudienceCS:2005 as defined in Annex A.11 of [TVA-Metadata]. When the IntendedAudienceCS is provided simultaneously with an instantiation of the ‘TargetUserProfile’, the two SHALL have equivalent meaning. • The network SHALL use the following URI syntax to signal terms from classification schemes: <classificationSchemeURI> “:” <termID> If this attribute is supported, terminals MAY also support the classification from [MIGFG]. In that case: • This classification SHALL be signalled with the URI “http://www.loc.gov/rr/mopic/miggen.html” • The value carried in the ‘Genre’ element SHALL be used to convey the actual value of the classification as given in [MIGFG] • The Network MAY use values “main” and “secondary” of the ‘type’ attribute so as to provide an ordering of two classification applying to the same Service. Other Classification Schemes MAY be signalled with the 'href' attribute, however how they are used is out of scope of this specification. For types 'main' and 'secondary', if this attribute is instantiated, the element ‘Genre’ SHALL be an empty string and the xml:lang attribute SHALL NOT be used. If this attribute is not instantiated, the ‘Genre’ element SHALL be a free string.	

Extension	E1	NM/ TM	0..N	Additional information related to this fragment. Contains the following attribute: url Contains the following element: Description	
url	A	NM/ TM	1	URL containing additional information related to this fragment.	anyURI
Description	E2	NM/ TM	0..N	Description regarding the additional information which can be retrieved from a web page. The language is expressed using built-in XML attribute 'xml:lang' with this element	string
				End of program guide	
PreviewData Reference	E1	NM/ TM	0..N	Reference to the 'PreviewData' fragment which specifies the preview data (eg. picture, video clip, or low-bit rate stream) associated with this service: It is possible that there are more than one 'PreviewDataReference' instances associated with the same fragment, in which case, the values of 'usage' attributes of these 'PreviewDataReference' instances SHALL be mutually exclusive. Contains the following attributes: idRef usage	
idRef	A	NM/ TM	1	Identification of the 'PreviewData' fragment which this fragment is associated with.	anyURI
usage	A	NM/ TM	1	Specifies the usage of the associated preview data. Possible values: 0. unspecified 1. Service-by-Service Switching 2. Service Guide Browsing 3. Service Preview 4. Barker 5. Alternative to blackout 6-127. reserved for future use 128-255. reserved for proprietary use The explanation and limitation on the above preview data usages is specified in section 5.7.	unsignedByte
BroadcastArea	E1	NO/ TO	0..1	Broadcast area to include location information for BCAST contents. Contains the following attribute: polarity Contains the following elements:	

				TargetArea hor_acc	
polarity	A	NO/ TO	0..1	Indication of whether the associated target area is intended for positive or negative terminal reception of the service. If polarity = true, this indicates the associated service is intended for reception by terminals located within the corresponding geographical area. (Default) If polarity = false, this indicates the associated service is not intended for reception by terminals located within the corresponding geographical area.	boolean
TargetArea	E2	NO/ TM	0..N	The target area to distribute contents (as specified in the [OMA MLP] with modifications) Contains the following elements: shape cc name_area ZipCode CellTargetArea	
shape	E3	NO/ TM	0..1	Shapes used to represent a geographic area that describes (as specified in the [OMA MLP])	
cc	E3	NO/ TM	0..1	Country code, 1-3 digits e.g. 355 for Albania (as specified in the [OMA MLP])	unsignedShort
name_area	E3	NO/ TM	0..N	Geopolitical name of area such as 'Seoul' (as specified in the [OMA MLP]. The instances of 'name_area' element differ only in language. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
ZipCode	E3	NO/ TM	0..1	Zip code	string
CellTargetArea	E3	NO/ TM	0..1	The target area to distribute content specified by he BDS specific service coverage area or minimum transmit area Contains the following attribute: type Contains the following element: CellArea	
type	A	NO/ TM	1	Allowed values are: 0 – Unspecified 1 - 3GPP Cell Global Identifier as defined in 3GPP TS 23.003 2 – 3GPP Routing Area Identifier (RAI) as defined in 3GPP TS 23.003 3 – 3GPP Location Area Identifier (LAI) as defined in 3GPP TS 23.003 4 – 3GPP Service Area Identifier (SAI) as defined in 3GPP TS 23.003 5 – 3GPP MBMS Service Area Identity (MBMS SAI) as defined in 3GPP TS 23.003	unsignedByte

				6 – 3GPP2 Subnet ID as defined in [3GPP2 C.S0054-0] 7 – 3GPP2 SID as defined in [3GPP2 C.S0005-D] 8 – 3GPP2 SID+NID as defined in [3GPP2 C.S0005-D] 9 – 3GPP2 SID+NID+PZID as defined in [3GPP2 C.S0005-D] 10 – 3GPP2 SID+PZID as defined in [3GPP2 C.S0005-D] 11 – DVB-H Cell ID (specified in section 6.3.4.1 of [BCAST10-DVBH-IPDC-Adaptation]) 12 - 127 reserved for future use 128 -255 reserved for proprietary use	
CellArea	E4	NO/ TM	0..N	The BDS specific transmit area given in the format as defined by type. Contains the following attribute: value Contains the following element: PP2CellID	
value	A	NO/TM	1	The value of the cell ID depending on the value of the type attribute	unsignedShort
PP2CellID	E5	NO/ TO	0..N	If type = 4, the value is Sector_ID as defined in [3GPP2 C.S0024-A] If type = 5, 6, 7 or 8, the value is BASE ID as defined in [3GPP2 C.S0002-0] 3GPP2 terminals SHALL support this element.	positiveInteger
hor_acc	E2	NO/ TM	0..N	Horizontal accuracy in meters (as specified in the [OMA MLP])	string
TermsOfUse	E1	NO/ TO	0..N	Element that declares there are Terms of Use associated with this fragment. Contains the textual presentation of Terms of Use or a reference to Terms of Use representation through 'PreviewData', and information whether user consent is required for the Terms of Use. Multiple occurrences of 'TermsOfUse' are allowed within this fragment, but for any two such occurrences values for elements 'Country' and 'Language' SHALL NOT be same at the same time. In addition to Terms of Use this element MAY be used for disclaimers, legal text and other pieces of information to be rendered to the user upon activation, purchase or consumption of service or content. Contains the following attributes: type id userConsentRequired Contains the following elements:	

				Country Language PreviewDataIDRef TermsOfUseText	
type	A	NM/ TM	1	The way the terminal SHALL interpret the Terms of Use: 0 – Display before purchasing or subscribing. If 'TermsOfUse' element of type '1' is present, terminal SHALL render the Terms of Use prior to initiating purchase or subscription request related PurchaseItem associated with this fragment. 1 – Display before playout. If 'TermsOfUse' element of type '2' is present, terminal SHALL present the Terms of Use prior to playing out content or service associated with this fragment. 2 - 127 reserved for future use 128 -255 reserved for proprietary use	unsignedByte
id	A	NM/ TM	1	The URI uniquely identifying the Terms of Use.	anyURI
userConsent Required	A	NM/ TM	1	Signals whether user consent for these Terms of Use is needed. true: User consent is required for these Terms of Use and needs to be confirmed in the subscription / purchase request message related to the PurchaseItem associated with this fragment. false: User consent is not required for the Terms of Use.	boolean
Country	E2	NM/ TM	1..N	List of countries for which the Terms of Use is applicable. Each value is a three character string according to ISO 3166-1 alpha-3	string
Language	E2	NM/ TM	1	Language in which the Terms of Use is given. Value is a three character string according to ISO 639-2 alpha standard for language codes.	string
PreviewData IDRef	E2	NO/ TM	0..N	Reference to the 'PreviewData' fragment which carries the representation of Terms of Use. If this element is not present, the 'TermsOfUseText' SHALL be present.	anyURI
TermsOfUse Text	E2	NO/ TO	0..1	Terms of Use text to be rendered. If 'PreviewDataIDRef' element is present under the 'TermsOfUse' this element SHALL NOT be present.	string
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.2 Schedule

The 'Schedule' fragment specifies the time when content item(s) of a service are made available for distribution and/or presentation. For scheduling purposes a set of content items can be associated with each other through 'Schedule' fragment. In such cases the detailed breakdown of scheduling information can be hidden from the user. The Schedule fragment also provides means for providing choices of consumption. These are defined in detail in sections 5.8. In addition, the following rules apply:

- If there are two or more 'Schedule' fragments referring to Content or 'Service' fragments overlapping, i.e., valid at the same *time*, then one of these 'Schedule' fragments SHALL be declared to be the default.
- The terminal A/V player SHALL rely on timing information declared by the Session Description information independent of what is announced in the 'Schedule' fragment.

Examples for the use of the relations between 'Service', 'Content' and 'Schedule' fragment are given in Appendix C.

The 'Schedule' fragment also can specify the distribution schedule and the automatic activation time of the InteractivityMediaDocuments associated within the 'InteractivityData' fragment which is referenced within this 'Schedule'.

Name	Type	Category	Cardinality	Description	Data Type
Schedule	E			'Schedule' fragment Contains the following attributes: id version defaultSchedule onDemand validFrom validTo Contains the following elements: ServiceReference InteractivityDataReference ContentReference PreviewDataReference TermsOfUse PrivateExt	
id	A	NM/ TM	1	ID of the 'Schedule' fragment. The the value of this attribute SHALL be globally unique."	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no 'validFrom' attribute is given.	unsignedInt
defaultSchedule	A	NM/ TM	0..1	Defines whether this schedule is the default schedule of the content or service item it relates, in case there are multiple schedules valid at the same time, see also section 5.8. If present, this field SHALL only be set to true.	boolean
onDemand	A	NM/TM	0..1	This attribute indicates the scheduled transmission of content that is accessible over the unicast channel where the transmission is not necessarily bounded into any specific interval of time. This type of content typically represents the "on-demand" component of a service.	boolean

				If this attribute is present and “true” it indicates that the referenced content item SHALL NOT be accessed automatically through this ‘Schedule’ fragment by the terminal when the associated service is selected and this ‘Schedule’ fragment is valid. This attribute SHALL NOT be present when this schedule fragment has no ContentReference element instantiated. Furthermore, this attribute in the 'Schedule' fragment MAY be set to ‘true’ if and only if the 'Access' fragments associated to the 'Schedule' fragment do not have the 'BroadcastServiceDelivery' element instantiated. See also section 5.8 If this attribute is present and “false” it indicates that the referenced content item MAY be accessed automatically through this ‘Schedule’ fragment by the terminal when the associated service is selected and this ‘Schedule’ fragment is valid, see section 5.8. The default value of this attribute is “false”.	
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
ServiceReference	E1	NM/ TM	1	Reference to the ‘Service’ fragment to which the ‘Schedule’ fragment belongs. If neither ‘InteractivityDataReference’ element nor ‘ContentReference’ element is present in the ‘Schedule’ fragment, then this ‘Schedule’ fragment defines the timeframe of the service availability as unbounded. Contains the following attributes: idRef audioLanguageIDRef textLanguageIDRef	
idRef	A	NM/ TM	1	Identification of the ‘Service’ fragment which this ‘Schedule’ fragment is associated with.	anyURI
audioLanguageIDRef	A	NM/ TM	0..1	This attribute selects one of the audio languages listed in the referred ‘Service’ fragment. See element ‘AudioLanguage’ and its attribute ‘id’ in the ‘Service’ fragment.	anyURI
textLanguageIDRef	A	NM/ TM	0..1	This attribute selects one of the text languages listed in the referred ‘Service’ fragment. See element ‘TextLanguage’ and its attribute ‘id’ in the ‘Service’ fragment.	anyURI
InteractivityDataReference	E1	NM/ TM	0..N	Identification of the ‘InteractivityData’ fragment to which the ‘Schedule’ fragment is associated.	

				Each occurrence of InteractivityDataReference implies the existence of a dedicated delivery schedule for those InteractivityMediaDocuments. This 'Schedule' fragment declares the available schedule for the file delivery of the InteractivityMediaDocuments referenced within the 'InteractivityData' fragment. Contains the following attributes: idRef Contains the following elements: AutoStart DistributionWindow Note: the presentation window is actually declared by the 'validFrom' and 'validTo' values in the InteractivityMediaDocument (see [BCAST10-Services] section 5.3.6).	
idRef	A	NM/ TM	1	Identification of the 'InteractivityData' fragment which the 'Schedule' fragment relates to.	anyURI
AutoStart	E2	NM/ TM	0..N	In the case the content or service the 'InteractivityData' fragment is related to is rendered, this element indicates the moment when the associated InteractivityMediaDocument is automatically activated if it is not rendered at that moment. If this element is not present or the terminal is turned on after 'AutoStart' time arrives, the associated InteractivityMediaDocument will not be automatically activated, but can be activated at any time of the validity of the media object document upon the user's request. It is RECOMMENDED that the terminal settings allow the users to configure whether to allow InteractivityMediaDocument to be automatically activated without users' request. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
Distribution Window	E2	NO/ TO	0..N	Time interval in which the referenced InteractivityMediaDocument referenced within the 'InteractivityData' fragment is available for delivery. Contains the following attributes: startTime endTime duration id	
startTime	A	NO/ TM	0..1	Start of DistributionWindow. If not given, the validity is assumed to have begun at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
endTime	A	NO/	0..1	End of DistributionWindow. If not given, the	unsignedInt

		TM		validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	
duration	A	NO/ TM	0..1	The maximum amount of time that the terminal which begins to acquire the Content item at the distribution endTime should allow to complete the acquisition. The unit of time is seconds.	unsignedInt
id	A	NO/ TO	0..1	Identification of the DistributionWindow which each DistributionWindow relates to. DistributionWindow id should be unique in the 'Schedule' fragment where it is declared.	unsignedInt
ContentReference	E1	NM/ TM	0..N	Reference to the 'Content' fragments that the 'Schedule' fragment relates to. Contains the following attributes: idRef contentLocation audioLanguageIDRef textLanguageIDRef repeatPlayback Contains the following elements: AutoStart DistributionWindow PresentationWindow	
idRef	A	NM/ TM	1	Identification of the 'Content' fragment which the 'Schedule' fragment relates to	anyURI
contentLocation	A	NM/ TM	0..1	Identifies the location of the associated content item in the file or stream delivery session described by the 'Access' fragment during the schedule defined by this fragment. When ALC is used for file delivery, this corresponds to the 'Content-Location' attribute' in the File element in the 'Access' fragment. When FLUTE is used for file delivery, this corresponds to the 'Content-Location' attribute in the FDT of the FLUTE session. When HTTP is used for file delivery, this corresponds to the 'Request-URI' to be used in the request line of HTTP request. When RTSP is used for stream delivery, this corresponds to the 'Request-URI' to be used in the request line of RTSP request. When the stream delivery is made available over broadcast access and no control protocol is necessary, this attribute SHALL NOT be used.	anyURI
audioLanguageIDRef	A	NM/ TM	0..1	This attribute selects one of the audio languages listed in the referred 'Content' fragment. See element 'AudioLanguage' and its attribute 'id' in the 'Content' fragment.	anyURI
textLanguageIDRef	A	NM/ TM	0..1	This attribute selects one of the text languages listed in the referred 'Content' fragment. See	anyURI

				element 'TextLanguage' and its attribute 'id' in the 'Content' fragment.	
repeatPlayback	A	NO/ TO	0..1	Indicates whether the content item referenced by the Presentation Window and/or Distribution Window in the 'Schedule' fragment is of the repeat playback type. Default: false	boolean
AutoStart	E2	NM/ TM	0..N	Indicates the moment when the associated content is automatically activated if it is not rendered at that moment. If this element is not present, the associated content will not be automatically activated, but can be activated at any time upon the user's request. It is RECOMMENDED that the terminal settings allow the users to configure whether to allow content to be automatically activated without users' request. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
Distribution Window	E2	NO/ TO	0..N	Time interval in which the referenced content specified by ContentID is available for delivery. Contains the following attributes: startTime endTime duration id The terminal SHALL support this element if it supports Cachecast and Cachecast related services	
startTime	A	NO/ TM	0..1	Start of DistributionWindow. If not given, the validity is assumed to have begun at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
endTime	A	NO/ TM	0..1	End of DistributionWindow. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
duration	A	NO/ TM	0..1	The maximum amount of time that the terminal which begins to acquire the Content item at the DistributionWindow 'endTime' should allow to complete the acquisition. The unit of time is in seconds	unsignedInt
id	A	NO/ TO	0..1	Identification of the DistributionWindow which the each DistributionWindow relates to. DistributionWindow id should be unique in the 'Schedule' fragment where it is declared.	unsignedInt
Presentation Window	E2	NM/ TM	0..N	Time interval in which the referenced content specified by ContentID is available for rendering. Contains the following attributes: startTime	

				endTime duration id	
startTime	A	NM/ TM	0..1	Start of PresentationWindow. If not given the validity is assumed is assumed to have begun at some time in the past. Note: When the content item is associated with the Cachecast service type, the startTime informs the user of the earliest instant he/she can begin the rendering of the associated content item. It does not represent the default start time at which the content is rendered by the terminal. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
endTime	A	NM/ TM	0..1	End of PresentationWindow. If not given, the validity is assumed to end in undefined time in the future. Note: When the content item is associated with the Cachecast service type, the endTime informs the user of the latest instant he/she can begin the rendering of the associated content item. It does not represent the default end time at which the content ceases to be rendered by the terminal. Rendering SHOULD run to completion. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
duration	A	NM/ TM	0..1	Time duration of the referenced content for rendering. It informs the user the latest startTime for which the content item can be rendered in its entirety. This attribute is REQUIRED when the content item is associated with the Cachecast service type.	unsignedInt
id	A	NM/ TM	0..1	Identification of the PresentationWindow PresentationWindow id should be unique in the 'Schedule' fragment where it is declared.	unsignedInt
PreviewData Reference	E1	NM/ TM	0..N	Reference to the 'PreviewData' fragment which specifies the preview data (eg. picture, video clip, or low-bit rate stream) associated with this schedule. The preview data associated with 'Schedule' fragment takes precedence over the preview data associated with the 'Content' fragments referenced by this 'Schedule' fragment. It is possible that there are more than one PreviewDataReference instances associated with the same fragment, in which case, the values of "usage" attributes of these PreviewDataReference instances SHALL be different. Contains the following attributes: idRef usage	
idRef	A	NM/ TM	1	Identification of the 'PreviewData' fragment which this fragment is associated with.	anyURI

usage	A	NM/ TM	1	Specifies the usage of the associated preview data. Possible values: 0. unspecified 1. Service-by-Service Switching 2. Service Guide Browsing 3. Service Preview 4. Barker 5. Alternative to blackout 6 -127. reserved for future use 128-255. reserved for proprietary use The explanation and limitation on the above preview data usages is specified in section 5.7.	unsignedByte
TermsOfUse	E1	NO/ TO	0..N	Element that declares there are Terms of Use associated with this fragment. Contains the textual presentation of Terms of Use or a reference to Terms of Use representation through 'PreviewData', and information whether user consent is required for the Terms of Use. Multiple occurrences of 'TermsOfUse' are allowed within this fragment, but for any two such occurrences values for elements 'Country' and 'Language' SHALL NOT be same at the same time. In addition to Terms of Use this element MAY be used for disclaimers, legal text and other pieces of information to be rendered to the user upon activation, purchase or consumption of service or content. Contains the following attributes: type id userConsentRequired Contains the following elements: Country Language PreviewDataIDRef TermsOfUseText	
type	A	NM/ TM	1	The way the terminal SHALL interpret the Terms of Use: 0 – Display before purchasing or subscribing. If 'TermsOfUse' element of type '1' is present, terminal SHALL render the Terms of Use prior to initiating purchase or subscription request related PurchaseItem associated with this fragment. 1– Display before playout. If 'TermsOfUse' element of type '2' is present, terminal SHALL present the Terms of Use prior to playing out content or service associated with this fragment.	unsignedByte

				2 - 127 reserved for future use 128 -255 reserved for proprietary use	
id	A	NM/ TM	1	The URI uniquely identifying the Terms of Use.	anyURI
userConsent Required	A	NM/ TM	1	Signals whether user consent for these Terms of Use is needed. true: User consent is required for these Terms of Use and needs to be confirmed in the subscription / purchase request message related to the PurchaseItem associated with this fragment. false: User consent is not required for the Terms of Use.	boolean
Country	E2	NM/ TM	1..N	List of countries for which the Terms of Use is applicable. Each value is a three character string according to ISO 3166-1 alpha-3	string
Language	E2	NM/ TM	1	Language in which the Terms of Use is given. Value is a three character string according to ISO 639-2 alpha standard for language codes.	string
PreviewData IDRef	E2	NO/ TM	0..N	Reference to the 'PreviewData' fragment which carries the representation of Terms of Use. If this element is not present, the 'TermsOfUseText' SHALL be present.	anyURI
TermsOfUse Text	E2	NO/ TM	0..1	Terms of Use text to be rendered. If 'PreviewDataIDRef' element is present under the 'TermsOfUse' this element SHALL NOT be present.	string
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.3 Content

'Content' fragment describes a content or programme that is a part of a service that the content refers to. A 'Content' fragment can be composed of several (audiovisual) media streams originating from several sources which together form a logical entity. An example is a single TV show broadcasted in several audio languages. A 'Content' fragment contains the metadata to describe a content or programme, represented by the attributes and elements in the following table. In the case of 'ParentalRating', 'TargetUserProfile', 'Genre', and 'BroadcastArea', should one or more of these elements exist in both the 'Content' fragment and its associated 'Service' fragment, the values of those elements in the 'Content' fragment take precedence over its counterparts in 'Service' fragment.

Name	Type	Category	Cardinality	Description	Data Type
Content	E			'Content' fragment Contains the following attributes: id version validFrom	

				validTo globalContentID emergency serviceContentProtection baseCID Contains the following elements: ServiceReference ProtectionKeyID Name Description StartTime EndTime AudioLanguage TextLanguage Length ParentalRating TargetUserProfile Genre Extension PreviewDataReference BroadcastArea TermsOfUse PrivateExt	
id	A	NM/ TM	1	ID of the 'Content' fragment. The the value of this attribute SHALL be globally unique.	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no 'validFrom' attribute is given.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
globalContentID	A	NM/ TM	0..1	The globally unique identifier identifying the content that this 'Content' fragment describes.	anyURI
emergency	A	NO/ TO	0..1	When assigned with value 'true', specifies that this content is content of emergency nature. This attribute can be used for presentation purposes to users. It is RECOMMENDED that the Terminal processes the reception of the services or content of emergency nature with high priority, and highlights their availability to user. How to order the emergency service or content is out of the	boolean

				scope of the specification. The default value of this attribute is 'false'.	
serviceContentProtection	A	NO/ TO	0..1	Specifies if the content is encrypted (true) or not (false). This element will be used for presentation purpose to users. Default: false	boolean
baseCID	A	NO/ TO	0..1	For the DRM Profile, part of the Service or Program CID used to identify the corresponding asset within an OMA DRM 2.0 Rights Object. The Service or Program CID is obtained from the BaseCID as described in [BCAST10-ServContProt], section 5.5.1]. In case this element is present the terminal SHALL use this element to identify the corresponding asset within an OMA DRM 2.0 Rights Object, instead of the baseCID(s) of the 'Service' fragment(s) this 'Content' fragment is associated with. In case this 'Content' fragment contains a reference to a 'Service' fragment this element MAY be present when: - this 'Content' fragment is referenced by a 'PurchaseItem' fragment or when - a 'PurchaseItem' fragment references a 'Schedule' fragment that references this 'Content' fragment, to identify the corresponding asset within an OMA DRM 2.0 Rights Object, in case the network supports the DRM profile. In case this 'Content' fragment does not contain a reference to a 'Service' fragment this element SHALL be present when: - this 'Content' fragment is referenced by a 'PurchaseItem' fragment or when - a 'PurchaseItem' fragment references a 'Schedule' fragment that references this 'Content' fragment to identify the corresponding asset within an OMA DRM 2.0 Rights Object, in case the network supports the DRM profile. The network and terminal SHALL support this element in case the DRM Profile is supported [BCAST10-ServContProt]. Note: for uniqueness of the baseCID see Appendix H.	string
ServiceReference	E1	NM/ TM	0..N	Reference to the 'Service' fragment(s) to which the 'Content' fragment belongs. Contains the following attributes: idRef weight	
idRef	A	NM/ TM	1	Identification of the 'Service' fragment which this 'Content' fragment is associated with.	anyURI

weight	A	NM/ TM	0..1	Intended order of display of this 'Content' fragment relative to other 'Content' fragments belonging to the same service as presented to the end user. The order of display is by increasing Weight value (i.e., content with lowest Weight is displayed first). Default: 65535	unsignedShort
ProtectionKeyID	E1	NO/ TO	0..N	List of key identifiers needed to access protected content. This information allows the terminal to determine whether or not it has the correct key material to access services within a PurchaseItem. In a scenario where this fragment is shared among multiple service providers, different key identifiers from the different service providers to access this specific protected service/content may be mixed in this element and the terminal SHOULD be able to sort out the key identifiers associated with the terminal's affiliated service provider based on the Key Domain ID. How this is used is out of scope and is left to implementation. The network and terminal SHALL support this element in case the Smartcard Profile is supported [BCAST10-ServContProt]. The protection key identifiers to access a specific service or content item SHALL only be signalled in one of the following fragment types: 'Service', 'Content', 'PurchaseItem' or 'Access' fragments, but not mixed. Contains the following attribute: type	base64Binary
type	A	NM/TM	1	Type of ProtectionKeyID: 0: ProtectionKeyID = Key Domain ID concatenated with SEK/PEK ID, where both values are as used in the Smartcard Profile [BCAST10-ServContProt] 1-127 Reserved for future use 128-255 Reserved for proprietary use	unsignedByte
				Start of program guide The program guide elements of this fragment are grouped between the Start of program guide and end of program guide cells in this fragment. The program guide elements are for user interpretation. This enables the content creator to provide user readable information about the service. The terminal SHOULD use all declared program guide elements in this fragment for presentation to the end-user. The terminal MAY offer search, sort etc functionalities.	

				The Program Guide consists of the following elements: Name Description StartTime EndTime AudioLanguage TextLanguage Length ParentalRating TargetUserProfile Genre Extension	
Name	E1	NM/ TM	1..N	Name of the 'Content' fragment, possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
Description	E1	NM/ TM	0..N	Description, possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
StartTime	E1	NM/ TM	0..1	The start time of the content which is for presentation purposes to the end user , expressed in UTC, using 'dateTime' XML built-in datatype. This element is applicable for scheduled rendering of non-Cachecast content. For Cachecast content, the start time is defined by the 'startTime' attribute of the 'PresentationWindow' element in the 'Schedule' fragment.	dateTime
EndTime	E1	NM/ TM	0..1	The end time of the content which is for presentation purposes to the end user, expressed in UTC, using 'dateTime' XML built-in datatype. This element is applicable for scheduled rendering of non-Cachecast content. For Cachecast content, the end time is defined by the 'endTime' attribute of the 'PresentationWindow' element in the 'Schedule' fragment.	dateTime
AudioLanguage	E1	NM/ TM	0..N	This element declares for the end users that this content is available with an audio stream corresponding to the language represented by the value of this element. The textual value of this element can be made available for the end users in different languages. In such a case the language used to represent the value of this element is signalled using the built-in XML attribute 'xml:lang'. See section 7 Multi-language support. As the instances of this element are representing choices w.r.t. language for the end users, the network needs to make sure the choices are unambiguous. Therefore there are certain restrictions on the values of the attributes 'id' and 'xml:lang'. The former are defined below and for the latter they are as follows: If the network instantiates the 'AudioLanguage'	string

				the following applies: • The value of the 'id' attribute SHALL be the same for 'Audiolanguage' elements that declare the same audio stream, but are made available to the end-user in different textual languages (different 'xml:lang' attributes). • For the case above, only one 'Audiolanguage' element can have the 'xml:lang' attribute omitted. • A particular 'id' value SHALL not be used to declare more than one audio streams for this fragment. If the attribute 'id' is omitted then terminal SHALL interpret this language setting being applicable for all the consumption methods available for the 'Content' in question. The network SHALL instantiate this element multiple times if and only if each of the instances contains the 'id' attribute. Contains the following attributes: id languageSDPTag	
id	A	NM/ TM	0..1	Identifier of the element 'AudioLanguage'. This is used to identify and select specific audio language upon reference by a 'Schedule' fragment. See section 7 about 'Multi-language support' and attribute 'audioLanguageIDRef' in 'Schedule' fragment.	anyURI
languageSDPTag	A	NM/ TO	1	Identifier of the audio language described by the parent 'AudioLanguage' element as used in the media sections describing the audio track in a Session Description. • The 'languageSDPTag' SHALL be formatted according to the rules of [RFC 3066], for the described language. • Each 'AudioLanguage' element declaring the same audio stream SHALL have the same value of the 'languageSDPTag'.	string
TextLanguage	E1	NM/ TM	0..N	This element declares for the end user that the textual components of this content are available in the language represented by the value of this element. The textual components can be, for instance, a caption or a sub-title track. The textual value of this element can be made available for the end users in different languages. In such a case the language used to represent the value of this element is signalled using the built-in XML attribute 'xml:lang. See section 7 Multi-	string

				language support. The same rules and constraints as specified for the element 'AudioLanguage' of assigning and interpreting the attributes 'id',' languageSDPTag' and 'xml:lang' SHALL be applied for this element also. If the attribute 'id' is omitted then terminal SHALL interpret this language setting being applicable for all the consumption methods available for the 'Content' in question. The network SHALL instantiate this element multiple times if and only if each of the instances contains the 'id' attribute. Contains the following attributes: id languageSDPTag	
id	A	NM/ TM	0..1	Identifier of the element 'TextLanguage'. This is used to identify and select specific language upon reference by a 'Schedule' fragment. See section 7 about 'Multi-language support' and attribute 'textLanguageIDRef' in 'Schedule' fragment.	anyURI
languageSDPTag	A	NM/TO	1	Identifier of the text language described by the parent 'TextLanguage' element as used in the media sections describing the textual track in a Session Description.	string
Length	E1	NM/ TM	0..1	Duration of the A/V content declared	duration
ParentalRating	E1	NM/ TM	0..N	The rating level defining criteria parents may use to determine whether the associated item is suitable for access by children, defined according to the regulatory requirements of the service area. The parental rating level defined for 'Content' fragment overrides the rating level defined for the corresponding 'Service' fragment during the validity of the 'Schedule' fragment. If there are multiple content items associated with a 'Schedule' fragment with different parental ratings, then the one with the most restrictive parental rating overrides the others. Contains the following attribute: ratingSystem	string
ratingSystem	A	NO/ TO	0..1	Specifies the parental rating system in use, in which context the value of 'ParentalRating' element is semantically defined. Absence of this attribute means that the rating system is undefined.	string
TargetUserProfile	E1	NO/ TO	0..N	Profile attributes of the users whom the content is targeting at. The detailed personal attribute names and the corresponding values are specified by attributes of 'attributeName' and 'attributeValue'. Amongst the possible profile attribute names are age, gender, occupation, etc. (subject to	

				national/local rules & regulations, if present and as applicable regarding use of personal profiling information and personal data privacy). The extensible list of 'attributeName' and 'attributeValue' pairs for a particular content enables end user profile filtering and end user preference filtering of broadcast contents. The terminal SHOULD be able to support 'TargetUserProfile' element. The terminal behavior for interpreting and acting upon 'TargetUserProfile' is out of the scope. It is RECOMMENDED that use of 'TargetUserProfile' element is an "opt-in" capability for users. Terminal settings SHOULD allow users to configure whether to input their personal profile or preference and whether to allow broadcast content to be automatically filtered based on the users' personal attributes without users' request. Contains the following attributes: attributeName attributeValue	
attributeName	A	NM/ TM	1	Profile attribute name.	string
attributeValue	A	NM/ TM	1	Profile attribute value.	string
Genre	E1	NM/ TM	0..N	Classification of content associated with characteristic form (e.g. comedy, drama). The OMA BCAST Service Guide allows describing the format of the Genre element in the Service Guide in two ways: • The first way is to use a free string • The second way is to use the "href" attributes of the Genre element to convey the information in the form of a controlled vocabulary (classification scheme as defined in [TVA-Metadata] or classification list as defined in [MIGFG]). The built-in XML attribute xml:lang MAY be used with this element to express the language. The Network MAY instantiate several different sets of 'Genre' element, using it as a free string or with a 'href' attribute. The Network SHALL ensure the different sets have equivalent and non-conflicting meaning, and the terminal SHALL select one of the sets to interpret for the end-user. Contains the following attributes: type href	string
type	A	NO/	0..1	This attribute signals the level of this 'Genre'	string

		TO		element. The following values are allowed: “main” “secondary” “other”	
href	A	NO/ TO	0..1	This attribute signals the controlled vocabulary used for this ‘Genre’ element. If this attribute is supported, terminals MAY support the following classification schemes according to [TVA-Metadata]: • for values of the ‘type’ attribute equal to "main" or "secondary", the terminal MAY support levels 1-4 of the TV Anytime ContentCS classification scheme identified by the classificationSchemeURI urn:tva:metadata:cs:ContentCS:2005 as defined in Annex A.8 of [TVA-Metadata] • for a value of the ‘type’ attribute equal to "other", the terminal MAY support levels 1-3 of the TV Anytime IntendedAudienceCS classification scheme identified by the classificationSchemeURI urn:tva:metadata:cs:IntendedAudienceCS:2005 as defined in Annex A.11 of [TVA-Metadata]. When the IntendedAudienceCS is provided simultaneously with an instantiation of the ‘TargetUserProfile’, the two SHALL have equivalent meaning. • The network SHALL use the following URI syntax to signal terms from classification schemes: <classificationSchemeURI> “:” <termID> If this attribute is supported, terminals MAY also support the classification from [MIGFG]. In that case: • This classification SHALL be signalled with the URI “http://www.loc.gov/rr/mopic/miggen.html” • The value carried in the ‘Genre’ element SHALL be used to convey the actual value of the classification as given in [MIGFG] • The Network MAY use values “main” and “secondary” of the ‘type’ attribute so as to provide an ordering of two classification applying to the same Service. Other Classification Schemes MAY be signalled	anyURI

				with the 'href' attribute, however how they are used is out of scope of this specification. For types 'main' and 'secondary', if this attribute is instantiated, the element ‘Genre’ SHALL be an empty string and the xml:lang attribute SHALL NOT be used. If this attribute is not instantiated, the ‘Genre’ element SHALL be a free string.	
Extension	E1	NM/ TM	0..N	Additional information related to this fragment. Contains the following attribute: url Contains the following element: Description	
url	A	NM/ TM	1	URL containing additional information related to this fragment.	anyURI
Description	E2	NM/TM	0..N	Description regarding the additional information which can be retrieved from a web page. The language is expressed using built-in XML attribute ‘xml:lang’ with this element	string
				End of program guide	
PreviewData Reference	E1	NM/ TM	0..N	Reference to the ‘PreviewData’ fragment which specifies the preview data (eg. picture, video clip, or low-bit rate stream) associated with this content. It is possible that there are more than one PreviewDataReference instances associated with the same fragment, in which case, the values of "usage" attributes of these PreviewDataReference instances SHALL be different. Contains the following attributes: idRef usage	
idRef	A	NM/ TM	1	Identification of the ‘PreviewData’ fragment which this fragment is associated with.	anyURI
usage	A	NM/ TM	1	Specifies the usage of the associated preview data. Possible values: 0. unspecified 1. Service-by-Service Switching 2. Service Guide Browsing 3. Service Preview 4. Barker 5. Alternative to blackout 6-127. reserved for future use 128-255. reserved for proprietary use The explanation and limitation on the above preview data usages is specified in section 5.7.	unsignedByte
BroadcastArea	E1	NO/ TO	0..1	Broadcast area to include location information for BCAST contents Contains the following attribute:	

				polarity Contains the following elements: TargetArea hor_acc	
polarity	A	NO/ TO	0..1	Indication of whether the associated target area is intended for positive or negative terminal reception of the content item. If polarity = true, this indicates the associated content item is intended for reception by terminals located within the corresponding geographical area. (Default) If polarity = false, this indicates the associated content item is not intended for reception by terminals located within the corresponding geographical area.	boolean
TargetArea	E2	NO/ TM	0..N	The target area to distribute contents (as specified in the [OMA MLP] with modifications) Contains the following elements: shape cc name_area ZipCode CellTargetArea	
shape	E3	NO/ TM	0..1	Shapes used to represent a geographic area that describes (as specified in the [OMA MLP])	
cc	E3	NO/ TM	0..1	Country code, 1-3 digits e.g. 355 for Albania (as specified in the [OMA MLP])	unsignedShort
name_area	E3	NO/ TM	0..N	Geopolitical name of area such as 'Seoul' (as specified in the [OMA MLP]). The instances of 'name_area' element differ only in language. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
ZipCode	E3	NO/ TM	0..1	Zip code	string
CellTargetArea	E3	NO/ TM	0..1	The target area to distribute content specified by he BDS specific service coverage area or minimum transmit area Contains the following attribute: type Contains the following element: CellArea	
type	A	NO/ TM	1	Allowed values are: 0 – Unspecified 1 - 3GPP Cell Global Identifier as defined in 3GPP TS 23.003 2 – 3GPP Routing Area Identifier (RAI) as defined in 3GPP TS 23.003 3 – 3GPP Location Area Identifier (LAI) as defined in 3GPP TS 23.003 4 – 3GPP Service Area Identifier (SAI) as defined	**unsignedByte**

				in 3GPP TS 23.003 5 – 3GPP MBMS Service Area Identity (MBMS SAI) as defined in 3GPP TS 23.003 6 – 3GPP2 Subnet ID as defined in [3GPP2 C.S0054-0] 7 – 3GPP2 SID as defined in [3GPP2 C.S0005-D] 8 – 3GPP2 SID+NID as defined in [3GPP2 C.S0005-D] 9 – 3GPP2 SID+NID+PZID as defined in [3GPP2 C.S0005-D] 10 – 3GPP2 SID+PZID as defined in [3GPP2 C.S0005-D] 11 – DVB-H Cell ID (specified in section 6.3.4.1 of [BCAST10-DVBH-IPDC-Adaptation]) 10 - 127 reserved for future use 128-255 reserved for proprietary use	
CellArea	E3	NO/ TM	0..N	The BDS specific transmit area given in the format as defined by type. Contains the following attribute: Value Contains the following element: PP2CellID	
value	A	NO/TM	1	The value of the cell ID depending on the value of the type attribute.	unsignedShort
PP2CellID	E4	NO/ TO	0..N	If type = 4, the value is Sector_ID as defined in [3GPP2 C.S0024-A] If type = 5, 6, 7 or 8, the value is BASE ID as defined in [3GPP2 C.S0002-0] Note: See relevant BDS specification for the data type of this element Note: 3GPP2 terminals SHALL support this element	positiveInteger
hor_acc	E2	NO/ TM	0..N	Horizontal accuracy in meters (as specified in the [OMA MLP])	string
TermsOfUse	E1	NO/ TO	0..N	Element that declares there are Terms of Use associated with this fragment. Contains the textual presentation of Terms of Use or a reference to Terms of Use representation through 'PreviewData', and information whether user consent is required for the Terms of Use. Multiple occurrences of 'TermsOfUse' are allowed within this fragment, but for any two such occurrences values for elements 'Country' and 'Language' SHALL NOT be same at the same time. In addition to Terms of Use this element MAY be used for disclaimers, legal text and other pieces of information to be rendered to the user upon activation, purchase or consumption of service or content.	

				Contains the following attributes: type id userConsentRequired Contains the following elements: Country Language PreviewDataIDRef TermsOfUseText	
type	A	NM/ TM	1	The way the terminal SHALL interpret the Terms of Use: 0 – Display before purchasing or subscribing. If 'TermsOfUse' element of type '1' is present, terminal SHALL render the Terms of Use prior to initiating purchase or subscription request related PurchaseItem associated with this fragment. 1 – Display before playout. If 'TermsOfUse' element of type '2' is present, terminal SHALL present the Terms of Use prior to playing out content or service associated with this fragment. 2 - 127 reserved for future use 128 -255 reserved for proprietary use	unsignedByte
id	A	NM/ TM	1	The URI uniquely identifying the Terms of Use.	anyURI
userConsentRequired	A	NM/ TM	1	Signals whether user consent for these Terms of Use is needed. true: User consent is required for these Terms of Use and needs to be confirmed in the subscription / purchase request message related to the PurchaseItem associated with this fragment. false: User consent is not required for the Terms of Use.	boolean
Country	E2	NM/ TM	1..N	List of countries for which the Terms of Use is applicable. Each value is a three character string according to ISO 3166-1 alpha-3	string
Language	E2	NM/ TM	1	Language in which the Terms of Use is given. Value is a three character string according to ISO 639-2 alpha standard for language codes.	string
PreviewDataIDRef	E2	NO/ TM	0..N	Reference to the 'PreviewData' fragment which carries the representation of Terms of Use. If this element is not present, the 'TermsOfUseText' SHALL be present.	anyURI
TermsOfUseText	E2	NO/ TM	0..1	Terms of Use text to be rendered. If 'PreviewDataIDRef' element is present under the 'TermsOfUse' this element SHALL NOT be present.	string
PrivateExt	E1	NO/	0..1	An element serving as a container for proprietary	

		TO		or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.4 Access

An 'Access' fragment describes to the terminal how it can access a service or a schedule during the lifespan of the 'Access' fragment. If the service or content is protected, the fragment also contains service and content protection information.

Name	Type	Category	Cardinality	Description	Data Type
Access	E			'Access' fragment Contains the following attributes: id version validFrom validTo Contains the following elements: AccessType KeyManagementSystem EncryptionType ServiceReference ScheduleReference TerminalCapabilityRequirement BandwidthRequirement ServiceClass PreviewDataReference NotificationReception PrivateExt	
id	A	NM/ TM	1	ID of the 'Access' fragment. The the value of this attribute SHALL be globally unique.	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the validFrom attribute, or as soon as it has been received if no validFrom attribute is given.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
AccessType	E1	NM/ TM	1	Defines the type of access. Note: Either one of 'BroadcastServiceDelivery' or 'UnicastServiceDelivery' but not both SHALL be instantiated. Implementation in XML Schema should use <choice>. Contains the following elements: BroadcastServiceDelivery	

				UnicastServiceDelivery	
BroadcastServiceDelivery	E2	NM/ TM	0..1	This element is used for the indication of IP transmission. Contains the following elements: BDSType SessionDescription FileDescription	
BDSType	E3	NM/ TM	0..1	Identifier of the type of underlying distribution system that this 'Access' fragment relates to. Contains the following element: Type Version	
Type	E4	NM/ TM	0..1	Type of underlying BDS, possible values: 0. IPDC over DVB-H 1. 3GPP MBMS 2. 3GPP2 BCMCS 3 -127. reserved for future use 128 -255. reserved for proprietary use	unsignedByte
Version	E4	NM/ TM	0..N	Version of underlying BDS. For instance, possible values are Rel-6 or Rel-7 for MBMS and 1x or HRPD or Enhanced HRPD for BCMCS.	string
SessionDescription	E3	NM/ TM	1	Reference to or inline copy of a Session Description information associated with this 'Access' fragment that the media application in the terminal uses to access the service. Note: a referenced 'SessionDescription' fragment may be delivered in two ways: via broadcast or via fetch over interaction channel. In the case of fetch over interaction channel, the 'SessionDescription' fragment can be acquired by accessing the URI (given as attribute of the different Session Description reference elements). Contains the following elements: SDP SDPRef USBDRef ADPRef The presence of elements 'SDP' and 'SDPRef' are mutually exclusive.	
SDP	E4	NM/ TM	0..1	An inlined Session Description in SDP format [RFC 4566]	string
SDPRef	E4	NM/TM	0..1	Reference to a Session Description in SDP format [RFC 4566] Contains the following attributes: uri idRef If both 'uri' and 'idRef' are present, the	

				referenced Session Description information SHALL be identical.	
uri	A	NM/ TM	0..1	The URI referencing an external resource containing SDP information.This URI is used for interactive retrieval.	anyURI
idRef	A	NM/ TM	0..1	The id of the referenced 'SessionDescription' fragment if the fragment is delivered over the broadcast channel in SGDU, globally unique	anyURI
USBDRef	E4	NM/TM	0..1	Reference to a Session Description in MBMS User Service Bundle Description as specified in [26.346] section 5.2.2, with the restrictions defined in section 5.1.2.5 of this spec. Contains the following attributes: uri idRef If both 'uri' and 'idRef' are present, the referenced Session Description information SHALL be identical.	
uri	A	NM/ TM	0..1	The URI referencing an external resource containing MBMS-USBD information.This URI is used for interactive retrieval.	anyURI
idRef	A	NM/ TM	0..1	The id of the referenced 'SessionDescription' fragment if the fragment is delivered over the broadcast channel in SGDU, globally unique	anyURI
ADPRef	E4	NM/TM	0..1	Reference to an AssociatedDeliveryProcedure for File and Stream Distribution as specified in [BCAST10-Distribution] section 5.3.4. Contains the following attributes: uri idRef If both 'uri' and 'idRef' are present, the referenced Session Description information SHALL be identical.	
uri	A	NM/ TM	0..1	The URI referencing an external resource containing AssociatedDeliveryProcedure for File and Stream Distribution. This URI is used for interactive retrieval.	anyURI
idRef	A	NM/ TM	0..1	The id of the referenced 'SessionDescription' fragment if the fragment is delivered over the broadcast channel in SGDU, globally unique	anyURI
FileDescription	E3	NO/ TM	0..1	File metadata for file delivery sessions. This element SHALL be provided when ALC is used. This element SHALL NOT be used in conjunction with FLUTE. The network SHALL support 'FileDescription' element and all its sub-elements and attributes if ALC is used for File Distribution function. Contains the following attributes: Content-Type	

				Content-Encoding FEC-OTI-FEC-Encoding-ID FEC-OTI-FEC-Instance-ID FEC-OTI-Maximum-Source-Block-Length FEC-OTI-Encoding-Symbol-Length FEC-OTI-Max-Number-of-Encoding-Symbols FEC-OTI-Scheme-Specific-Info Contains the following elements: File	
Content-Type	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	string
Content-Encoding	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	string
FEC-OTI-FEC-Encoding-ID	A	NO/TM	0..1	See RFC 3926, section 3.4.2	unsignedByte
FEC-OTI-FEC-Instance-ID	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Maximum-Source-Block-Length	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Encoding-Symbol-Length	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Max-Number-of-Encoding-Symbols	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Scheme-Specific-Info	A	NO/TM	0..1	This attribute MAY be used to communicate FEC information which is not adequately represented by the other attributes related to FEC.	base64Binary
File	E4	NO/ TM	1..N	Parameters of a file. Contains the following attributes: Content-Location TOI Content-Length Transfer-Length Content-Type Content-Encoding Content-MD5 FEC-OTI-FEC-Encoding-ID FEC-OTI-FEC-Instance-ID FEC-OTI-Maximum-Source-Block-Length FEC-OTI-Encoding-Symbol-Length FEC-OTI-Max-Number-of-Encoding-Symbols	

				FEC-OTI-Scheme-Specific-Info	
Content-Location	A	NO/ TM	1	See RFC 3926, section 3.4.2	anyURI
TOI	A	NO/ TM	1	See RFC 3926, section 3.4.2	positiveInteger
Content-Length	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
Transfer-Length	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
Content-Type	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	string
Content-Encoding	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	string
Content-MD5	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	base64Binary
FEC-OTI-FEC-Encoding-ID	A	NO/TM	0..1	See RFC 3926, section 3.4.2	unsignedByte
FEC-OTI-FEC-Instance-ID	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Maximum-Source-Block-Length	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Encoding-Symbol-Length	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Max-Number-of-Encoding-Symbols	A	NO/ TM	0..1	See RFC 3926, section 3.4.2	unsignedLong
FEC-OTI-Scheme-Specific-Info	A	NO/TM	0..1	This attribute MAY be used to communicate FEC information which is not adequately represented by the other attributes related to FEC.	base64Binary
UnicastServiceDelivery	E2	NM/ TM	0..N	This element indicates which server and/or protocol is used for delivery of service over Interaction Channel. Contains the following attribute: type Contains the following elements: AccessServerURL SessionDescription ServiceAccessNotificationURL	
type	A	NM/ TM	1	Specifies transport mechanism that is used for this access. 0 - HTTP 1 - WAP 1.0 2- WAP 2.x	unsignedByte

				3- Generic RTSP to initialize RTP delivery 4- RTSP to initialize RTP delivery as per 3GPP-PSS (3GPP packet-switched streaming service) 5- RTSP to initialize RTP delivery as per 3GPP2-MSS (3GPP2 multimedia streaming services) 6 - FLUTE over Unicast 7-127 Reserved for future use 128-255 Reserved for proprietary use Note: Specification or negotiation of ports used for unicast service delivery is handled by the used unicast distribution mechanisms. For example, RTSP and PSS based systems (values 3 and 4) do port negotiation within the RTSP signalling exchange.	
AccessServerURL	E3	NM/ TM	1..N	Server URL from which the terminal can receive the service via the Interaction Network as specified in section 5.5 and 6.7 of [BCAST10-Distribution]. For example, AccessServerURL can be an HTTP URL pointing to downloadable content, or an RTSP URL pointing to a streaming server for starting a streaming session.	anyURI
SessionDescription	E3	NM/ TM	1	Reference to or inline copy of a Session Description information associated with this 'Access' fragment that the media application in the terminal uses to access the service. Note: a referenced 'SessionDescription' fragment may be delivered in two ways: via broadcast or via fetch over interaction channel. In the case of fetch over interaction channel, the 'SessionDescription' fragment can be acquired by accessing the URI (given as attribute of the different Session Description reference elements). Contains the following elements: SDP SDPRef USBDRef ADPRef The presence of elements 'SDP' and 'SDPRef' are mutually exclusive.	
SDP	E4	NM/ TM	0..1	An inlined Session Description in SDP format [RFC 4566]	string
SDPRef	E4	NM/ TM	0..1	Reference to a Session Description in SDP format [RFC 4566] Contains the following attributes: uri idRef If both 'uri' and 'idRef' are present, the referenced Session Description information SHALL be identical.	

uri	A	NM/ TM	0..1	The URI referencing an external resource containing SDP information.This URI is used for interactive retrieval.	anyURI
idRef	A	NM/ TM	0..1	The id of the referenced 'SessionDescription' fragment if the fragment is delivered over the broadcast channel in SGDU, globally unique	anyURI
USBDRef	E4	NM/TM	0..1	Reference to a Session Description in MBMS User Service Bundle Description as specified in [26.346] section 5.2.2, with the restrictions defined in section 5.1.2.5 of this spec. Contains the following attributes: uri idRef If both 'uri' and 'idRef' are present, the referenced Session Description information SHALL be identical.	
uri	A	NM/ TM	0..1	The URI referencing an external resource containing MBMS-USBD information.This URI is used for interactive retrieval.	anyURI
idRef	A	NM/ TM	0..1	The id of the referenced 'SessionDescription' fragment if the fragment is delivered over the broadcast channel in SGDU, globally unique	anyURI
ADPRef	E4	NM/TM	0..1	Reference to an AssociatedDeliveryProcedure for File and Stream Distribution as specified in [BCAST10-Distribution] section 5.3.4. Contains the following attributes: uri idRef If both 'uri' and 'idRef' are present, the referenced Session Description information SHALL be identical.	
uri	A	NM/ TM	0..1	The URI referencing an external resource containing AssociatedDeliveryProcedure for File and Stream Distribution. This URI is used for interactive retrieval.	anyURI
idRef	A	NM/ TM	0..1	The id of the referenced 'SessionDescription' fragment if the fragment is delivered over the broadcast channel in SGDU, globally unique	anyURI
ServiceAccessNotificationURL	E3	NM/ TM	0..N	URL that the terminal SHOULD use to notify the BSD/A when it accesses (switches to) this service over this unicast access. The 'ServiceAccessNotificationURL' MAY be used in conjunction with 'UnicastServiceDelivery' types 3, 4, 5 or 6. If used, the device SHOULD NOT use RTSP TEARDOWN and RTSP SETUP to terminate an existing RTSP stream and set up a new one. The terminal SHALL NOT use this URL for notification without user consent. Note: This URL can for example be used for initiating server-managed channel switching in	anyURI

				unicast transmission.	
KeyManagementSystem	E1	NM/ TM	0..N	Information of Key Management System(s)(KMS) that can be used to contact the BCAST Permissions Issuer and, in case of the SmartCard Profile whereby GBA is used for SMK derivation, whether GBA_U is mandatory or whether either GBA_ME or GBA_U can be used. Note that the BCAST Permissions Issuer can support more than one KMS. If KeyManagementSystem is not specified, it means no service or content protection is applied. Contains the following elements: ProtectionKeyID PermissionsIssuerURI TerminalBindingKeyID Contains the following attributes: kmsType protectionType	
kmsType	A	NM/ TM	1	Identifies the type of Key Management System(s)(KMS). Possible values: 0. oma-bcast-drm-pki Indicates OMA BCAST DRM profile (Public Key Infrastructure) 1. oma-bcast-gba_u-mbms Indicates BCAST Smartcard profile using GBA_U (Symmetric Key Infrastructure) 2. oma-bcast-gba_me-mbms Indicates BCAST Smartcard profile using GBA_ME 3. oma-bcast-prov-bcmcs Indicates provisioned 3GPP2 BCMCS SKI 4 -127 Reserved for future use 128 – 255 Reserved for proprietary use	unsignedByte
protectionType	A	NM/ TM	1	Specifies the protection type offered by the KMS. Values: 0. Content protection only, as defined by the LTKM (protection_after_reception in STKM = 0x00 or 0x01 [BCAST10-ServContProt]) 1. Service protection only (protection_after_reception in STKM = 0x03 [BCAST10-ServContProt]) 2. Content protection as defined by LTKM, plus playback of protected recording permission (protection_after_reception in STKM = 0x02 [BCAST10-ServContProt]) 3 – 127 Reserved for future use 128 – 255 Reserved for proprietary use	unsignedByte

				This attribute may also be used for presentation purpose to users, to indicate whether the content item(s), associated with the referenced Schedule by this 'Access' fragment, is protected or not.	
PermissionsIssuerURI	E2	NM/TM	1	The address of the BCAST Permissions Issuer to which requests for key material, tokens and /or consumption rules should be sent by the BCAST Terminal. Contains the following attribute: type	anyURI
type	A	NM/TM	1	The type of the PermissionsIssuerURI, identified by the following values: false– DRM Profile true – Smartcard Profile Note: In the case of the DRM Profile, the PermissionsIssuerURI corresponds to the RightsIssuerURL as defined by [DRMDRM-v2.0]. In the case of the Smartcard Profile, the PermissionsIssuerURI corresponds to the network entity (i.e. the BSM) to which all BCAST Service Provisioning messages are sent by the terminal.	boolean
ProtectionKeyID	E2	NO/ TO	0..N	List of key identifiers needed to access protected content. This information allows the terminal to determine whether or not it has the correct key material to access services within a PurchaseItem. In a scenario where this fragment is shared among multiple service providers, different key identifiers from the different service providers to access this specific protected service/content may be mixed in this element and the terminal SHOULD be able to sort out the key identifiers associated with the terminal's affiliated service provider based on the Key Domain ID. How this is used is out of scope and is left to implementation. The network and terminal SHALL support this element in case the Smartcard Profile is supported [BCAST10-ServContProt]. The protection key identifiers to access a specific service or content item SHALL only be signalled in one of the following fragment types: 'Service', 'Content', 'PurchaseItem' or 'Access' fragments, but not mixed. Contains the following attribute: type	base64Binary
type	A	NM/TM	1	Type of ProtectionKeyID: 0: ProtectionKeyID = Key Domain ID concatenated with SEK/PEK ID, where both values are as used in the Smartcard Profile [BCAST10-ServContProt]. 1-127 Reserved for future use 128-255 Reserved for proprietary use	unsignedByte

TerminalBindingKeyID	E2	NO/ TO	0..1	Number identifying the Terminal Binding Key ID (TBK ID) that is needed to access the service. If the element is absent, no TerminalBindingKey is used. Both the network and the terminal with the Smartcard Profile SHALL support this element. It is TM for terminals with the smartcard profile. This element does not apply to the DRM profile. Contains the following attribute: tbkPermissionsIssuerURI	unsignedInt
tbkPermissionsIssuerURI	A	NO/ TM	0..1	Specifies the Permissions Issuer URI for the TerminalBindingKey if it is different from the 'PermissionsIssuerURI' sub-element of the 'KeyManagementSystem' element. If the attribute is not present the same 'PermissionsIssuerURI' indicated for KeyManagementSystem is used to acquire both SEK / PEK and TerminalBindingKey.	anyURI
EncryptionType	E1	NM/ TM	0..N	Specifies which encryption methods the terminal is to be able to support in order to utilize this Access. Contains the same value as traffic_protection_protocol signalled in STKM. 0 – IPsec 1 – STRP 2 – ISMACryp 3 – DCF 4- 255 – Reserved for future use. If this element is not present, this Access is not encrypted.	unsignedByte
ServiceReference	E1	NM/ TM	0..N	Reference to the 'Service' fragment(s) to which the 'Access' fragment belongs. Either one of 'ServiceReference' or 'ScheduleReference', or neither, but not both SHALL be instantiated. Each 'Service' fragment SHALL be associated to at least one 'Access' fragment to enable the terminal to access the Service. A single 'Access' fragment MAY reference to multiple 'Service' fragments. This is used when there are several independent descriptions of a single Service.	
idRef	A	NM/ TM	1	Identification of the 'Service' fragment which this 'Access' fragment is associated with.	anyURI
ScheduleReference	E1	NM/ TM	0..N	Reference to the 'Schedule' fragment(s) to which the 'Access' fragment belongs. This provides a reference to a 'Schedule' fragment to temporarily override the default 'Access' fragment of the Service addressed by the Schedule. Either one of 'ServiceReference' or	

				'ScheduleReference', or neither, but not both SHALL be instantiated. Note: Implementation in XML Schema using <choice>. Contains the following attribute: idRef Contains the following element: DistributionWindowID	
idRef	A	NM/ TM	1	Identification of the 'Schedule' fragment which the 'Access' fragment relates to.	anyURI
Distribution WindowID	E2	NO/ TM	0..N	Relation reference to the DistributionWindowID to which the 'Access' fragment belongs. The 'DistributionWindowID' element declared in this element SHALL be the complete collection or a subset of the DistributionWindow ids declared in the 'Schedule' fragment, to which 'idRef' reference belongs.	unsignedInt
TerminalCapabilityRequirement	E1	NO/ TM	0..1	Terminal capabilities needed to consume the service or content. This element provides a hint to the terminal of what is needed to apply to consumption method represented by this 'Access' fragment. It is out of scope of this specification how the terminal applies this information. For video and audio, the media type and the related 'type' attribute in the SDP (see section 5.1.2.5) signal the audio/video decoder. This way, these parameters complement the TerminalCapabilityRequirement. Additionally, the complexities of the audio/video streams are described here if they differ from the complexities which can be derived from the media type attributes in the SDP (e.g. level). In this case, the parameters defined in the 'Access' fragment take priority. Contains the following elements: Video Audio DownloadFile	
Video	E2	NO/ TM	0..1	Video codec capability related requirements Contains the following elements: Complexity	
Complexity	E3	NO/ TM	1	The complexity the video decoder has to deal with. It is RECOMMENDED that this element is included if the complexity indicated by the MIME type parameters in the SDP differs from the actual complexity. Contains the following elements: Bitrate Resolution	

				MinimumBufferSize	
Bitrate	E4	NO/ TM	0..1	The total bit-rate of the video stream. Contains the following attributes: average maximum	
average	A	NO/ TM	0..1	The average bit-rate in kbit/s	unsignedShort
maximum	A	NO/ TM	0..1	The maximum bit-rate in kbit/s	unsignedShort
Resolution	E4	NO/ TM	0..1	The resolution of the video. Contains the following attributes: horizontal vertical temporal	
horizontal	A	NO/ TM	1	The horizontal resolution of the video in pixels.	unsignedShort
vertical	A	NO/ TM	1	The vertical resolution of the video in pixels.	unsignedShort
temporal	A	NO/ TM	1	The maximum temporal resolution in frames per second.	decimal
MinimumBufferSize	E4	NO/ TM	0..1	The minimum decoder buffer size needed to process the video content in kbytes.	unsignedInt
Audio	E2	NO/ TM	0..1	The audio codec capability. Contains the following element: Complexity	
Complexity	E3	NO/ TM	1	The complexity the audio decoder has to deal with. It is RECOMMENDED that this element is included if the complexity indicated by the MIME type parameters in the SDP differs from the actual complexity. Contains the following elements: Bitrate MinimumBufferSize	
Bitrate	E4	NO/ TM	0..1	The total bit-rate of the audio stream. Contains the following attributes: average maximum	
average	A	NO/ TM	0..1	The average bit-rate in kbit/s	unsignedShort
maximum	A	NO/ TM	0..1	The maximum bit-rate in kbit/s	unsignedShort
MinimumBufferSize	E4	NO/ TM	0..1	The minimum decoder buffer size needed to process the audio content in kbytes.	unsignedInt
DownloadFil	E2	NO/	0..1	The required capability for the download files.	

e		TM		Contains the following elements: MIMEType	
MIMEType	E3	NO/ TM	1..N	Assuming a download service consists of a set of files with different MIME types which together make up the service, the terminal must support all of these MIME types in order to be able to present the service to the user. The format of this string • SHALL follow the 'Content-Type' syntax defined in [RFC 2045]. • Additionally the 'Content-Type' MAY be augmented as defined in [RFC 4281]. In the latter case the 'Content-Type' SHALL begin by • "audio/3gpp", • "audio/3gpp2", • "video/3gpp" , • "video/3gpp2" Contains the following attribute: codec	string
codec	A	NO/ TM	0..1	The codec parameters for the associated MIME Media type. If the file's MIME type definition specifies mandatory parameters, these MUST be included in this string. Optional parameters containing information that can be used to determine as to whether the terminal can make use of the file SHOULD be included in the string. One example of the parameters defined for audio/3GPP, audio/3GPP2, video/3GPP, video/3GPP2 is specified in [RFC4281].	string
BandwidthRequirement	E1	NO/ TM	0..1	Specification of needed network bandwidth in kbit/s to the access channel described in this fragment. A broadcast service can include multiple accessible streams (same content) with different bandwidth, so that the terminal can make a choice depending on its current reception condition.	unsignedInt
ServiceClass	E1	NM/ TM	1..N	The ServiceClass identifies the class of service. This identification is more detailed than the 'ServiceType' element in the 'Service' fragment and allows the association of service / access combination to specific applications. The Terminal SHALL be able to interpret the ServiceClasses as defined in OMNA registry (see Appendix E).	string
PreviewData Reference	E1	NM/ TM	0..N	Reference to the 'PreviewData' fragment which specifies the preview data (eg. picture, video clip, or low-bit rate stream).associated with this access. It is possible that there are more than one PreviewDataReference instances associated with	

				the same fragment, in which case, the values of "usage" attributes of these PreviewDataReference instances SHALL be different. Contains the following attributes: idRef usage	
idRef	A	NM/ TM	1	Identification of the 'PreviewData' fragment which this fragment associated with.	anyURI
usage	A	NM/ TM	1	Specifies the usage of the associated preview data. Possible values: 0. unspecified 1. Service-by-Service Switching 2. Service Guide Browsing 3. Service Preview 4. Barker 5. Alternative to blackout 6-127. reserved for future use 128-255. reserved for proprietary use The explanation and limitation on the above preview data usages is specified in section 5.7.	unsignedByte
Notification Reception	E1	NM/ TM	0..1	Reception information for service-specific Notification Messages. In case of delivery over Broadcast channel, 'IPBroadcastDelivery' specifies the address information for receiving Notification message. In case of delivery over Interaction channel, 'RequestURL' specifies address information for subscribing notification, 'PollURL' specifies address information for polling notification. If this element is present, at least one of the elements "IPBroadcastDelivery", "RequestURL", or "PollURL" SHALL be present. Contains the following elements: IPBroadcastDelivery RequestURL PollURL	
IPBroadcast Delivery	E2	NM/TM	0..1	Provides IP multicast address and port number for reception of Notification Messages over the broadcast channel. The 'port' is MANDATORY in both Network and Terminal because a designated UDP Port has to be used to deliver the Notification Message through an on-going session or the designated session while the 'address' is OPTIONAL to be used for the delivery of Notification Messages through the designated multicast or broadcast session. In case the 'address' attribute is not provided the destination address provided by this access fragment SHALL be assumed. Contains the following attributes:	

				port address	
port	A	NM/ TM	1	Service-specific Notification Message delivery UDP destination port number, delivery over broadcast channel.	unsignedInt
address	A	NM/ TM	0..1	Service-specific Notification Message delivery IP multicast address, delivery over broadcast channel.	string
RequestURL	E2	NM/ TM	0..1	URL through which the terminal can subscribe to service-specific Notification Messages.	anyURI
PollURL	E2	NM/ TM	0..1	URL through which the terminal can poll service-specific Notification Messages.	anyURI
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.5 Session Description Information

The Session Description information MAY be directly contained in the associated 'Access' fragment. Further, such information, in the form of 'SessionDescription' fragments MAY be referenced by 'idRef' attribute in the associated 'Access' fragment, and encapsulated into a SGDU for broadcast delivery. Finally, 'SessionDescription' fragments MAY be referenced from the associated 'Access' fragment by the 'uri' attribute containing an absolute URI pointing to an external resource which is fetched via the interaction channel.

5.1.2.5.1 SessionDescription fragment

An 'Access' fragment MAY reference zero, one or more 'SessionDescription' fragments by the 'SessionDescriptionReference' element in the following manner.

- The 'Access' fragment MAY reference zero or more 'SessionDescription' fragments, containing Session Description information, formatted according to the syntax of Session Description Protocol (SDP) [RFC 4566], or
- The 'Access' fragment MAY reference zero or more 'SessionDescription' fragments containing MBMS User Service Bundle Descriptions (USBD) as specified in [26.346] section 5.2.2, or
- The 'Access' fragment MAY reference zero or more 'SessionDescription' fragments containing associated delivery procedure description information as specified in [BCAST10-Distribution] section 5.3.4., formatted in XML.

The following applies to the SDP-formatted 'SessionDescription' fragment.

- For IPv6 support in SDP, RFC 3266 is used.
- For IPv4 support in SDP, RFC 4566 is used.
- 'SessionDescription' fragments MAY also contain other SDP extensions.
- If the SDP contains multiple media sections with the same media identifier (e.g. m=audio... or m=video...), then the selection between the media section to be used by the terminal is out of scope of this specification.
- When several SDP are made available, representing different audio and/or subtitle instances of a given service or content, the Network SHALL ensure that all of the said SDP declare exactly the same video media description. This enables terminals not supporting dynamic SDP updates to merge all SDP declarations of a given service or content into one internally generated SDP to feed the multimedia player. How this is done by the terminal is out of the scope of this specification. However, it is emphasized that the terminal:
  - can use the 'languageSDPTag' attribute from the 'AudioLanguage' and 'TextLanguage' elements in 'Service' fragment and/or 'Content' fragment to provide the correct language tag upon merging an audio or subtitle media declaration into the internally generated SDP.

- might need to rename security related stream declaration upon merging them into the internally generated SDP, and rearrange their linkage to the media declaration they apply to.

The Session Description information SHALL be provided using either syntax of SDP in text format, or through a 3GPP MBMS User Service Bundle Description (USBD) [3GPP TS 26.346]. BCAST device SHALL support SDP and MAY support USBD. The MBMS USBD as used in BCAST SHALL contain zero or one reference to FEC Repair Stream Description and SHALL contain one UserServiceDescription (USD). Each USD SHALL refer to one or more DeliveryMethods (DM), and SHALL contain zero or more accessGroup elements (those containing accessBearer elements). Each DM SHALL refer to zero or one AssociatedDeliveryProcedureDescriptions, and SHALL refer to one SessionDescription, and SHALL NOT refer to any SecurityDescription.

5.1.2.5.2 Session Description for broadcast streamed media session

The SessionDescription SHALL provide the following parameters:

- Destination IP address and port number for each media in the session
- The start time and end time of the session
- The transport protocol used
- Media types and media formats
- Data rates using SDP bandwidth modifiers

Additionally, the Session Description MAY provide the following parameters:

- The sender IP address
- The mode of MBMS bearer per media
- FEC configuration and related parameters
- Initial buffering delay, using the 'min-buffer-time' attribute as specified in [ETSI 102 472] section 5.3.4.
- Service protection parameters as defined in [BCAST10-ServContProt]

For the above parameteres, either SDP [RFC 4566] or 'SessionDescription' fragments in MBMS user service description of MBMS User Service Bundle Description (MBMS-USBD) [26.346], or [ETSI 102 472] section 5.2 SHALL be used to describe a broadcast streamed media session. An example is given below.

Each media line (m=<media> <port> <transport> <fmt list>) in the SDP indicates a payload type which is expressed using a <fmt list> sub-field. If the type definition specifies mandatory parameters, these MUST be included in the payload type associated attributes. Optional parameters containing information that can be used to determine as to whether the decoder installed in the terminal can decode the stream/content SHOULD be included in the payload type associated attributes

SDP Example:

```
v=0
o=- 424 3292855200 IN IP6 FF15:0:0:0:0:0:81:1BC
s=Unencrypted Mobile TV Example
c=IN IP6 FF15:0:0:0:0:0:81:1BD
t=0 0
m=audio 49172 RTP/AVP 96
b=AS:64
a=rtpmap:96 mpeg4-generic/32000
a=fmtp:96 streamtype=5; profile-level-id=15; mode=AAC-hbr; config=1290; SizeLength=13;IndexLength=3;
IndexDeltaLength=3; Profile=1;
m=video 49170 RTP/AVP 97
b=AS:250
a=rtpmap:97 H264/90000
a=fmtp:97 profile-level-id=42c00d; packetization-mode=1;sprop-parameter-sets=Z0LADZtAoPiA,aN4liA==;
```

5.1.2.5.3 Session Description for broadcast file delivery session

The ALC specification [RFC 3450] describes required and optional parameters for an ALC session and media descriptions. The FLUTE specification [RFC 3926] also states a set of required and optional parameters for describing a FLUTE session. This section specifies the Session Description using SDP for both ALC and FLUTE sessions that is used for file distribution sessions. The formal specification of the parameters is given in ABNF [RFC 2234].

5.1.2.5.3.1. Session Descriptors for FLUTE Sessions

The Session Description for FLUTE sessions SHALL provide the following parameters:

- Sender IP address
- The number of channels in the session
- Destination IP address and port number for each channel in the session, given in the corresponding media line
- The Transport Session Identifier of the session
- The file delivery protocol ID
- The start time and end time of the session

Additionally, the Session Description MAY provide the following parameters:

- The mode of MBMS bearer per media
- FEC configuration and related parameters
- Data rates using SDP bandwidth modifiers

The parameters SHALL be formatted according to the definitions in section 7.3 of [3GPP TS 26.346] or section 6.1.13 of [ETSI 102 472].

The Session Description MAY be referenced in an MBMS User Service Bundle Description (MBMS-USBD) [26.346].

Example of FLUTE Session Description:

```
v=0
o=user123 2890844526 2890842807 IN IP6 2201:056D::112E:144A:1E24
s= Example of file delivery session description using FLUTE
i=More information
t=2873397496 2873404696
a=FEC-declaration:0 encoding-id=0;
a=source-filter: incl IN IP6 * 2001:210:1:2:240:96FF:FE25:8EC9
a=flute-tsi:3
a=flute-ch:1
m=application 12345 FLUTE/UDP 0
c=IN IP6 FF1E:03AD::7F2E:172A:1E24/1
b=AS:64
a=FEC:0
```

5.1.2.5.3.2. Session Descriptors for ALC Sessions

The session parameters defined for FLUTE sessions SHALL also apply for ALC sessions. For these parameters, SDP [RFC 4566] SHALL be used.

The Session Description MAY be referenced in an MBMS User Service Bundle Description (MBMS-USBD) [26.346].

The parameters for ALC sessions SHALL be formatted according to the following syntax and semantics.

Sender IP address

There SHALL be exactly one IP sender address per file distribution session, and thus there SHALL be exactly one IP source address per complete file distribution session SDP description. The source IP address SHALL be provided using a 'source-filter' attribute, which has the following syntax in ABNF:

source-filter="a=source-filter: incl IN" SP addr-type SP dest-address SP src-list

addr-type= "IP4" | "IP6"

dest-address="*"

src-list = unicast-address

unicast-address is an IP4 or IP6 address depending on addr-type.

The following exceptions apply to the source-filter:

- Exactly one source address MAY be specified by this attribute such that exactly one source address is given by the src-list field.
- There SHALL be exactly one 'source-filter' attribute per complete file distribution session SDP description, and this SHALL be in the session part of the Session Description (i.e., not per media).

Note that the destination address is given as "*", which indicates that the source filter applies to all destination addresses.

Number of channels

Multiple channels MAY be used for several purposes. Receivers that are capable of receiving multiple channels simultaneously can benefit from the delivery of urgent files over multiple channels to receive the file at a faster rate. Multiple channels can also be used to deliver files with different FEC encodings, so that receivers can select the FEC code that they support to receive the file.

The multiple channel attribute parameter indicates to the receiver the number of channels the sender is using in the ALC session to transmit data. The value specified by this descriptor MAY be used by the receiver to check consistency of the SDP by counting the number of *m*-lines describing the destinations.

The number of channels is given in SDP syntax for ALC sessions as follows:

alc-channel-line = "a=alc-ch:" ch CRLF

ch = integer; integer is defined in [ABNF]

where ch is the number of channels used by the sender to transmit data in a file distribution session.

Destination IP Address and Port Number per Channel, and Media and Format List

Each channel SHALL be described by the media-level channel descriptor. These channel parameters SHALL be per channel:

- IP destination address
- Destination port number.

The IP destination address SHALL be defined according to the "connection data" field ("c=") of SDP [RFC 4566]. The destination port number SHALL be defined according to the <port> sub-field of the media announcement field ("m=") of SDP. The media announcement field is specified in [RFC 4566] as follows:

```
m=media SP port ["/" integer] SP proto 1*(SP fmt) CRLF
```

The protocol identifier, which is given by the proto sub-field, SHALL be ALC/UDP for ALC sessions. The media sub-field SHALL be set to "application" and the format list SHALL be set to "0" to indicate that formats are not in use. Only one port SHALL be defined per media-line, so that there SHALL be exactly one media-line per channel.

The presence of an ALC session on a certain channel SHALL be indicated by using the '*m*-line' in the SDP description as shown in the following example:

m=application 12345 ALC/UDP 0

c=IN IP6 FF1E:03AD::7F2E:172A:1E24

Note that the above destination address is an IPv6 multicast address.

Transport Session Identifier (TSI)

The combination of the TSI and the IP source address identifies the ALC session. Each TSI SHALL uniquely identify an ALC session for a given IP source address during the time that the session is active and also for a large enough time before and after the active session time.

The TSI SHALL be defined according to the SDP attribute given below. There SHALL be exactly one occurrence of this descriptor in a complete SDP Session Description and it SHALL appear at session level.

The syntax for an ALC session is given below in ABNF format:

alc-tsi-line = “a=alc-tsi:” tsi CRLF

tsi=1*DIGIT

Where *tsi* gives the TSI.

Session Timing Parameters

A file distribution session start and end times SHALL be defined according to the SDP timing field (“t=”) [RFC 4566].

FEC capabilities and related parameters

A FEC-declaration session-level attribute is defined which results in, e.g.:

a=FEC-declaration:0 encoding-id=0

Several FEC-declaration lines MAY be declared for the same session. The FEC-declaration is OPTIONAL as the information may be available elsewhere (e.g. in the codepoint header field of LCT header). If this attribute is not used and is not available elsewhere, the terminal SHALL assume that support for FEC encoding id 0 (Compact No-Code FEC code) is sufficient capability to enter the session.

A new media-level FEC-reference attribute ‘FEC‘ SHALL be defined to refer to the used FEC declaration. It can result in, e.g.;

a=FEC:0

This attribute is used as a short hand to inherit one of one or more session-level FEC-declarations to a specific media (i.e. channel).

The syntax for the attributes in ABNF [18] is:

fec-declaration-line = "a=FEC-declaration:" fec-ref SP fec-enc-id [";" SP fec-inst-id] CRLF

fec-ref = 1*3DIGIT (value is the SDP-internal identifier for FEC-declaration).

fec-enc-id = "encoding-id=" enc-id

enc-id = 1*DIGIT (value is the FEC Encoding ID used).

fec-inst-id = "instance-id=" inst-id

inst-id = 1*DIGIT (value is the FEC Instance ID used).

fec-line = "a=FEC:" fec-ref CRLF

The SDP declares the default FEC encoding scheme (on session level). The FEC encoding scheme may however change from file to file and this is overwritten by declarations in the EXT_FTI ALC/LCT header, or in File Description of the Service Guide.

Bandwidth Specification

The maximum bit-rate occupied by each channel of the ALC session SHALL be specified using the "AS" bandwidth modifier [RFC 4566] on media level (i.e. for each single channel). The Application Specific (AS) maximum bandwidth for an ALC channel SHALL be the largest sum of the sizes of all packets transmitted during any one second long period, expressed in kilobits. The size of the packet SHALL be the complete packet including the protocol headers, i.e. IP, UDP, ALC/LCT headers and the data payload.

Example of ALC Session Description

```
v=0
o=user123 2890844526 2890842807 IN IP6 2201:056D::112E:144A:1E24
s=Example of file delivery session description using ALC
i=More information
t=2873397496 2873404696
a=FEC-declaration:0 encoding-id=0;
a=FEC-declaration:1 encoding-id=1;
a=source-filter: incl IN IP6 *2201:056D::112E:144A:1E24
a=alc-tsi:3
a=alc-ch :2
m=application 12345 ALC/UDP 0
c=IN IP6 FF1E:03AD::7F2E:172A:1E24
b=AS:64
a=FEC:0
m=application 12346 ALC/UDP 0
c=IN IP6 FF1E:03AD::7F2E:172A:1E25
b=AS:64
a=FEC:1
```

5.1.2.6 Purchase Item

A purchase item groups one or multiple services/contents, or schedules associated with specific services or contents that an end-user can purchase or subscribe to as a whole.

An instance of the ‘PurchaseItem’ fragment SHALL only reference one type of ‘Service’, ‘Schedule’, ‘Content’, or ‘PurchaseItem’ fragments. This constraint is expressed by the using the ‘choice’ element in the Service Guide XML schema.

Name	Type	Category	Cardinality	Description	Data Type
PurchaseItem	E			‘PurchaseItem’ fragment Contains the following attributes: id version validFrom validTo globalPurchaseItemID binaryPurchaseItemID weight closed Contains the following elements: ServiceReference	

				ScheduleReference ContentReference PurchaseItemReference ProtectionKeyID Name Description StartTime EndTime Extension ParentalRating Dependencies Exclusions PrivateExt	
id	A	NM/ TM	1	ID of the 'PurchaseItem' fragment. The the value of this attribute SHALL be globally unique.	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no validFrom attribute is given.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp. The validFrom time of the PurchaseItem SHALL be no earlier than the latest of the validFrom time(s) of the referenced PurchaseItem(s).	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp. The 'validTo' time of the PurchaseItem SHALL be no later than the earliest of the validTo time(s) of the referenced PurchaseItem(s).	unsignedInt
globalPurchaseItemID	A	NM/TM	1	The globally unique identifier identifying the purchase item described by this fragment. The 'globalPurchaseItemID' is used when identifying the purchase item during purchase or other purchase item related actions (see [BCAST10-Services]).	anyURI
binaryPurchaseItemID	A	NO/TO	0..1	A 32-bit identifier generated by BSM, identifying the purchase item described by this fragment. The binaryPurchaseitemID is used to relate this purchase item with the corresponding BCRO. Networks and Broadcast Devices implementing the DRM Profile [BCAST10-ServContProt] SHALL support this attribute.	unsignedInt
weight	A	NM/TM	0..1	Intended order of display of this purchase item relative to other purchase items as seen by the end user. The order of display is by increasing weight value (i.e., purchase item with lowest weight is displayed first). Default: 65535	unsignedShort

closed	A	NM/TM	0..1	If present and value = true, it indicates the Purchase Item is closed to new subscribers. If value = false, unspecified. Default: false.	boolean
ServiceReference	E1	NM/ TM	0..N	References to the 'Service' fragments which belong to this PurchaseItem. Note: a 'Service' fragment can be referenced by multiple PurchaseItems. Contains the following attribute: idRef	
idRef	A	NM/ TM	1	Identification of the 'Service' fragment which this 'PurchaseItem' fragment is associated with.	anyURI
ScheduleReference	E1	NM/ TM	0..N	References to the 'Schedule' fragments which belong to this PurchaseItem. Note: a 'Schedule' fragment can be referenced by multiple 'PurchaseItem' fragments. Contains the following attribute: idRef Contains the following element: PresentationWindowIDRef	
idRef	A	NM/TM	1	Identification of the 'Schedule' fragment which the 'PurchaseItem' fragment relates to.	anyURI
PresentationWindowIDRef	E2	NM/TM	0..N	Relation reference to the PresentationWindow to which the 'PurchaseItem' fragment belongs. The 'PresentationWindowIDRef' declared in this element SHALL be the complete collection or a subset of the PresentationWindow ids declared in the 'Schedule' fragment, to which the above Schedule 'idRef' reference belongs.	unsignedInt
ContentReference	E1	NM/ TM	0..N	References to the 'Content' fragments which belong to this PurchaseItem. Contains the following attribute: idRef Note: a 'Content' fragment can be referenced by multiple 'PurchaseItem' fragments.	
idRef	A	NM/ TM	1	Identification of the 'Content' fragment which this 'PurchaseItem' fragment is associated with.	anyURI
PurchaseItemReference	E1	NM/TM	0..N	References to the 'PurchaseItem' fragments that are included in the purchase or subscription of this PurchaseItem, and thus need not be purchased separately. Note: a 'PurchaseItem' fragment can be referenced by multiple PurchaseItems. The depth of the 'PurchaseItem' tree SHALL NOT be more than three. 'PurchaseItem' fragment referencing other 'PurchaseItem' fragments SHALL NOT cause contradictory or circular reference chains. The reference to a PurchaseItem SHALL NOT	

				depend on, nor be excluded by, the subscription to other purchaseable PurchaseItem(s), if the latter PurchaseItem(s) are indeed not subscribed. Contains the the following attribute: idRef	
idRef	A	NM/ TM	1	Identification of the 'PurchaseItem' fragment which this 'PurchaseItem' fragment is associated with.	anyURI
ProtectionKeyID	E1	NO/TO	0..N	List of key identifiers needed to access protected content. This information allows the terminal to determine whether or not it has the correct key material to access services within a PurchaseItem. In a scenario where this fragment is shared among multiple service providers, different key identifiers from the different service providers to access this specific protected service/content may be mixed in this element and the terminal SHOULD be able to sort out the key identifiers associated with the terminal's affiliated service provider based on the Key Domain ID. How this is used is out of scope and is left to implementation. The network and terminal SHALL support this element in case the Smartcard Profile is supported [BCAST10-ServContProt]. The protection key identifiers to access a specific service or content item SHALL only be signalled in one of the following fragment types: 'Service', 'Content', 'PurchaseItem' or 'Access' fragments, but not mixed. Contains the following attribute: type	base64Binary
type	A	NM/TM	1	Type of ProtectionKeyID, possible values: 0: ProtectionKeyID = Key Domain ID concatenated with SEK/PEK ID, where both values are as used in the Smartcard Profile [BCAST10-ServContProt] 1-127 Reserved for future use 128-255 Reserved for proprietary use	unsignedByte
				Start of program guide The program guide elements of this fragment are grouped between the Start of program guide and end of program guide cells in this fragment. The program guide elements are for user interpretation. This enables the content creator to provide user readable information about the service. The terminal SHOULD use all declared program guide elements in this fragment for presentation to the end-user. The terminal MAY offer search, sort etc functionalities.	

				The Program Guide consists of the following elements: Name Description StartTime EndTime ParentalRating Extension	
Name	E1	NM/ TM	1..N	Name of the PurchaseItem, possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
Description	E1	NM/TM	0..N	Description of the purchase item, possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
StartTime	E1	NM/TM	0..1	The first moment when the item can be purchased, which is for presentation purposes to the end user in UTC, using 'dateTime' XML built-in datatype. yyyy=year mm=month dd=day T = time separator to clarify the different use of 'mm' hh=hour mm = minutes	dateTime
EndTime	E1	NM/TM	0..1	The last moment when the item can be purchased, which is for presentation purposes to the end user in UTC, using 'dateTime' XML built-in datatype. yyyy=year mm=month dd=day T = time separator to clarify the different use of 'mm' hh=hour mm = minutes	dateTime
ParentalRating	E1	NM/ TM	0..N	The rating level defining criteria parents can use to determine whether the associated item is suitable for access by children, defined according to the regulatory requirements of the service area This determines the rating level for service purchase, not the rating level of the actual service consumption. Contains the following attribute: ratingSystem	string
ratingSystem	A	NO/TO	0..1	Specifies the parental rating system in use, in which context the value of 'ParentalRating' element is semantically defined. Absence of this attribute means that the rating system is undefined.	string
Extension	E1	NM/ TM	0..N	Additional information related to this fragment. Contains the following attribute: url	

				Contains the following element: Description	
url	A	NM/ TM	1	URL containing additional information related to this fragment.	anyURI
Description	E2	NM/TM	0..N	Description regarding the additional information which can be retrieved from a web page. The language is expressed using built-in XML attribute 'xml:lang' with this element	string
				End of program guide	
Dependency Reference	E1	NO/TO	0..N	Reference to those 'PurchaseItem' fragments, subscription to at least one of which is REQUIRED before subscription SHOULD be offered for this PurchaseItem. The depth of the PurchaseItem tree SHALL NOT be more than three. 'PurchaseItem' fragment referencing other 'PurchaseItem' fragments SHALL NOT cause contradictory or circular reference chains. Contains the the following attribute: idRef	
idRef	A	NM/TM	1	Identification of the 'PurchaseItem' fragment that is referenced by this element.	anyURI
ExclusionReference	E1	NO/TO	0..N	Reference to those 'PurchaseItem' fragments, subscription to which SHOULD NOT be offered after the user subscribes to this PurchaseItem. The depth of the PurchaseItem tree SHALL NOT be more than three. 'PurchaseItem' fragment referencing other 'PurchaseItem' fragments SHALL NOT cause contradictory or circular reference chains. Contains the following attribute: idRef	
idRef	A	NM/TM	1	Identification of the 'PurchaseItem' fragment that is referenced by this element.	anyURI
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.7 Purchase Data

The Purchase Data fragment serves the following purposes:

- To express from which purchase channel a certain purchase item can be purchased
- To express all information about a purchase item that is specific for the purchase channel (which can include the price information)

Name	Type	Category	Cardinality	Description	Data Type
PurchaseData	E			'PurchaseData' fragment Contains the following attributes:	

				id version validFrom validTo Contains the following elements: Description PriceInfo PromotionInfo Extension PurchaseItemReference PurchaseChannelReference PreviwDataReference TermsOfUse PrivateExt	
id	A	NM/ TM	1	ID of the 'PurchaseData' fragment. The the value of this attribute SHALL be globally unique.	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no 'validFrom' attribute is given.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
				Start of program guide The program guide elements in this fragment are grouped between the Start of program guide and end of program guide cells in this fragment. The program guide elements are for user interpretation. This enables the content creator to provide user readable information about the service. The terminal SHOULD use all declared program guide elements in this fragment for presentation to the end-user. The terminal MAY offer search, sort etc functionalities. The Program Guide consists of the following elements: Description PriceInfo PromotionInfo Extension	
Description	E1	NM/ TM	0..N	Description of the purchase data, possibly in multiple languages. The language is expressed	string

				using built-in XML attribute 'xml:lang' with this element.	
PriceInfo	E1	NM/ TM	0..1	Specifies the price information of the purchase item associated with this 'PurchaseData' fragment If the price is not given, it will be negotiated with the user as part of the purchase transaction. In this case, the 'PurchaseData' fragment merely reflects that a certain purchase item can be purchased from the purchase channel. Contains the following attribute: subscriptionType Contains the following elements: MonetaryPrice TotalNumberToken SubscriptionPeriod	
subscription Type	A	NO/ TM	1	The type of subscription offered. Possible values: 0 – one-time subscription: the subscription will last for as long as the 'SubscriptionPeriod' element indicates, and the user will be charged for the amount indicated by the 'MonetaryPrice' element. 1 – open-ended subscription: the subscription will be valid until the user unsubscribes. In this case the subscription period, if present, indicates the frequency at which the user will be charged for the amount specified by the Price element. 2-127 Reserved for future use 128-255 Reserved for proprietary use	unsignedByte
MonetaryPrice	E2	NM/ TM	0..N	Specifies the monetary value of the price for subscribing to the associated purchase item if 'SubscriptionPeriod' is present, or the monetary value of a token package if 'TotalNumberToken' is present. Only one 'MonetaryPrice' per currency SHALL be defined. Contains the following attribute: currency	decimal
currency	A	NM/ TM	1	Specifies the monetary currency codes defined in ISO 4217 international currency codes.	string
TotalNumberToken	E2	NM/ TM	0..1	Specifies the number of tokens which can be acquired as a token package for consumption of the associated purchase item. Note: the value of 'MonetaryPrice' divided by the value of 'TotalNumberToken' is the cost per token, the terminal can calculate and display this information to the user if necessary. Contains the following attributes: tokenType	unsignedShort

				consumptionAmount consumptionUnit maxReplay When 'tokenType'=2 or 3, the value of this element divided by the value of 'consumptionAmount' represents the number of tokens consumed per instance of play. When 'tokenType'=4 or 5, the value of this element divided by the value of 'consumptionAmount' represents the number of tokens consumed per time unit of play.	
tokenType	A	NM/ TM	1	Specifies the type of tokens. Possible values are: 0 – unspecified 1 – tokens for the DRM Profile 2 – time tokens for service purse of Smartcard Profile 3 – time tokens for user purse of Smartcard Profile 4 – play tokens for service purse of Smartcard Profile 5 – play tokens for user purse of Smartcard Profile 6-127 – reserved for future use 128-255 – reserved for proprietary use Note: type 1 tokens are applicable only to DRM Profile, whereas types 2-5 tokens are applicable only to Smartcard Profile. Time tokens are deducted according to length of time of content consumption (i.e. PPT) whereas play tokens are deducted based on number of plays of content (i.e. PPV). For a definition of user purse and service purse, see section 6.6.2.2 and section 6.6.5 of [BCAST10-ServContProt].	unsigneByte
consumption Amount	A	NM/TM	0..1	Represents the amount of permitted consumption corresponding to this token package. When 'consumptionUnit' = 0 or 1 or 2, this provides the total view time corresponding to the total number of tokens specified by 'TotalNumberToken'. When 'consumptionUnit' = 3, this provides the total number of plays or views corresponding to the total number of tokens specified by 'TotalNumberToken'. This attribute SHALL be present when the amount of permitted consumption corresponding to this token package is a fixed number. When that is not the case, due to variable cost per play instance or per time unit, as expressed in the LTKM for a given SEK / PEK ID, this attribute SHALL NOT be present.	unsignedSho rt
consumption Unit	A	NM/TM	1	Describes the type of consumption unit. Allowed values are:	unsignedByt e

				0 – time in seconds, used for 'tokenType' = 2 or 3 1 – time in minutes, used for 'tokenType' = 2 or 3 2 – time in hours, used for 'tokenType' = 2 or 3 3 – number of plays, used for 'tokenType' = 4 or 5 4 -127 reserved for future use 128-255 reserved for proprietary use	
maxReplay	A	NM/ TM	0..1	This attribute indicates the maximum number or time of plays for this purchase item, as an integer multiple of 'consumptionAmount'. It informs the user of the practical limit on the number of token packages that can be requested in the Token Purchase Request message (as indicated by the value of 'purchaseUnitNum') defined in section 5.1.5.5.1 of [BCAST10-Services]. How this is managed by the BSM is out of scope. If this attribute is not present, it means there is no limitation on number or time of plays.	unsignedShort
Subscription Period	E2	NM/ TM	0..1	Specifies the time duration of the subscription period of the purchase item referenced by this 'PurchaseData' fragment. e.g. 1 year as P1Y, 2 month as P2M, 1 hour as P1H, etc. This element is only applicable for subscription-based consumption, not applicable for PPV/PPT based consumption.	duration
PromotionInfo	E1	NO/ TO	0..N	Information of the promotion activities/coupons related to the PurchaseItem. If 'TargetUserProfile' sub-element is present, then PromotionInfo can be displayed if the profile of the terminal user meets the requirements described by 'TargetUserProfile'. Contains the following attributes: id validFrom validTo Contains the following elements: Title TargetUserProfile Description PromotionExtension	
id	A	NO/ TM	1	Identifier of one certain PromotionInfo, unique for BSM. This id MAY be used in the purchase process to identify the specific promotion	unsignedInt
validFrom	A	NO/ TM	0..1	Start of validity; if not given, the start of validity is assumed in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validTo	A	NO/ TM	0..1	End of validity; if not given, the end of validity is assumed in the distant future, and the end time can be specified later by updating the object. This field contains the 32bits integer part of an NTP	unsignedInt

				time stamp.	
Title	E2	NO/ TM	1..N	Title of the PromotionInfo. The instances of Title element differ only in language. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
TargetUserProfile	E2	NO/ TO	0..N	Profile attributes of the users whom the information of the promotion activity/coupon is targeting at. The detailed personal attribute names and the corresponding values are specified by attributes of 'attributeName' and 'attributeValue'. Amongst the possible profile attribute names are age, gender, occupation, etc. (subject to national/local rules & regulations, if present and as applicable regarding use of personal profiling information and personal data privacy.). The extensible list of 'attributeName' and 'attributeValue' pairs for a particular PromotionInfo enables end user profile filtering and end user preference filtering of PromotionInfo for the PurchaseData. The terminal SHOULD be able to support 'TargetUserProfile' element. The terminal behavior for interpreting and acting upon 'TargetUserProfile' is out of the scope of this document. It is RECOMMENDED that use of 'TargetUserProfile' element is an "opt-in" capability for users. Terminal settings SHOULD allow users to configure whether to input their personal profile or preference and whether to allow PromotionInfo to be automatically filtered based on the users' personal attributes without users' request. Contains the following attributes: attributeName attributeValue	
attributeName	A	NO/ TM	1	Profile attribute name.	string
attributeValue	A	NO/ TM	1	Profile attribute value.	string
Description	E2	NO/ TM	0..N	Description or explanation about the PromotionInfo. The language is expressed using the built-in XML attribute 'xml:lang' with this element. Either 'Description' or 'PromotionExtension' element or both of them SHOULD be specified by the BSM to represent the detailed information on this PromotionInfo.	string
PromotionExtension	E2	NO/ TM	0..N	Additional detailed promotional information (e.g. information about coupon sponsors, server location for purchases by using coupons). Either 'Description' or 'PromotionExtension'	

				element or both of them SHOULD be specified by the BSM to represent the detailed information on this PromotionInfo. Contains the following attribute: url Contains the following element: Description	
url	A	NO/ TM	1	URL containing additional information related to this promotion.	anyURI
Description	E3	NO/ TM	0..N	Description regarding the additional information which can be retrieved from a web page. The language is expressed using built-in XML attribute 'xml:lang' with this element	string
Extension	E1	NO/ TM	0..N	Additional information related to this fragment. Contains the following attribute: url Contains the following element: Description	
url	A	NO/ TM	1	URL containing additional information related to this fragment.	anyURI
Description	E2	NO /TM	0..N	Description regarding the additional information which can be retrieved from a web page. The language is expressed using built-in XML attribute 'xml:lang' with this element	string
				End of program guide	
PurchaseItemReference	E1	NM/ TM	1	The PurchaseItem to which this PurchaseData applies to. Contains the following attribute: idRef	
idRef	A	NM/ TM	1	Identification of the 'PurchaseItem' fragment which this 'PurchaseData' fragment is associated with.	anyURI
PurchaseChannelReference	E1	NM/ TM	1..N	The PurchaseChannel through which the identified PurchaseItem can be obtained. Contains the following attribute: idRef	
idRef	A	NM/ TM	1	Identification of the PurchasChannel fragment which this 'PurchaseData' fragment is associated with.	anyURI
PreviewData Reference	E1	NM/ TM	0..N	Reference to the 'PreviewData' fragment which specifies the preview data (eg. picture, video clip, or low-bit rate stream).associated with this purchase data. It is possible that there are more than one PreviewDataReference instances associated with the same fragment, in which case, the values of "usage" attributes of these PreviewDataReference instances SHALL be different.	

				Contains the following attributes: idRef usage	
idRef	A	NM/ TM	1	Identification of the 'PreviewData' fragment which this 'PurchaseData' fragment is associated with.	anyURI
usage	A	NM/ TM	1	Specifies the usage of the associated preview data. Possible values: 0. unspecified 1. Service-by-Service Switching 2. Service Guide Browsing 3. Service Preview 4. Barker 5. Alternative to blackout 6-127. reserved for future use 128-255. reserved for proprietary use The explanation and limitation on the above preview data usages is specified in section 5.7.	unsignedByte
TermsOfUse	E1	NO/ TO	0..N	Element that declares there are Terms of Use associated with this fragment. Contains the textual presentation of Terms of Use or a reference to Terms of Use representation through 'PreviewData', and information whether user consent is required for the Terms of Use. Multiple occurrences of 'TermsOfUse' are allowed within this fragment, but for any two such occurrences values for elements 'Country' and 'Language' SHALL NOT be same at the same time. In addition to Terms of Use this element MAY be used for disclaimers, legal text and other pieces of information to be rendered to the user upon activation, purchase or consumption of service or content. Contains the following attributes: type id userConsentRequired Contains the following elements: Country Language PreviewDataIDRef TermsOfUseText	
type	A	NM/ TM	1	The way the terminal SHALL interpret the Terms of Use: 0 – Display before purchasing or subscribing. If 'TermsOfUse' element of type '1' is present, terminal SHALL render the Terms of Use prior to initiating purchase or subscription request related	unsignedByte

				PurchaseItem associated with this fragment. 1 – Display before playout. If 'TermsOfUse' element of type '2' is present, terminal SHALL present the Terms of Use prior to playing out content or service associated with this fragment. 2 - 127 reserved for future use 128 -255 reserved for proprietary use	
id	A	NM/ TM	1	The URI uniquely identifying the Terms of Use.	anyURI
userConsent Required	A	NM/ TM	1	Signals whether user consent for these Terms of Use is needed. true: User consent is required for these Terms of Use and needs to be confirmed in the subscription / purchase request message related to the PurchaseItem associated with this fragment. false: User consent is not required for the Terms of Use.	boolean
Country	E2	NM/ TM	1..N	List of countries for which the Terms of Use is applicable. Each value is a three character string according to ISO 3166-1 alpha-3	string
Language	E2	NM/ TM	1	Language in which the Terms of Use is given. Value is a three character string according to ISO 639-2 alpha standard for language codes.	string
PreviewData IDRef	E2	NO/ TM	0..N	Reference to the 'PreviewData' fragment which carries the representation of Terms of Use. If this element is not present, the 'TermsOfUseText' SHALL be present.	anyURI
TermsOfUse Text	E2	NO/ TM	0..1	Terms of Use text to be rendered. If 'PreviewDataIDRef' element is present under the 'TermsOfUse' this element SHALL NOT be present.	string
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.8 Purchase Channel

The 'PurchaseChannel' fragment represents a system from which access and content rights can be purchased by the terminal (and its end-user).

Name	Type	Category	Cardinality	Description	Data Type
PurchaseChannel	E			'PurchaseChannel' fragment Contains the following attributes: id version validFrom	

				validTo rightsIssuerURI Contains the following elements: Name Description ContactInfo PortalURL PurchaseURL Extension PrivateExt	
id	A	NM/ TM	1	ID of the 'PurchaseChannel' fragment. The the value of this attribute SHALL be globally unique.	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no 'validFrom' attribute is given.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
rightsIssuer URI	A	NO/ TO	0..1	ID of the rights issuer associated with the BSM. This information is needed to allow unconnected devices to identify the Rights Issuer Services (specified in section 12 of [DRM20-Broadcast-Extensions]) that may be operated by their Home BSM. The network and terminal SHALL support this attribute when the DRM Profile with broadcast only mode [DRM20-Broadcast-Extensions] is supported.	anyURI
				Start of program guide The program guide elements of this fragment are grouped between the Start of program guide and end of program guide cells in this fragment. The program guide elements are for user interpretation. This enables the content creator to provide user readable information about the service. The terminal SHOULD use all declared program guide elements in this fragment for presentation to the end-user. The terminal MAY offer search, sort etc functionalities. The Program Guide consists of the following elements:	

				Name Description ContactInfo PortalURL PurchaseURL Extension	
Name	E1	NM/ TM	1..N	Name of the Purchase Channel, possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
Description	E1	NM/ TM	0..N	Description of the purchase channel, possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string
ContactInfo	E1	NM/ TM	0..1	A text string that indicates to a user how to contact a BSM to initiate an out-of-band purchase transaction (e.g. phone number, URL etc)	string
PortalURL	E1	NM/ TM	0..1	The URL on which the BSM may offer service related information and/or web-based service provisioning via HTTP or HTTPS. Contains the following attribute: supportedService	anyURI
supportedService	A	NM/ TM	0..1	Specifies how the Terminal is expected to use 'PortalURL' and 'PurchaseURL'. 0: The Terminal SHALL use the Service provisioning messages of [BCAST10-Services] to 'PurchaseURL' to enable the service provisioning functionality. Further, the Terminal MAY contact the 'PortalURL' via HTTP to acquire further information on the purchase items available on this purchase channel. 1: The Terminal SHALL be able to access the 'PortalURL' via HTTP to acquire further information on purchase items available on this purchase channel and to achieve service provisioning functionality. Further, the Terminal SHALL perform the service provisioning directly on the 'PortalURL' and SHALL NOT send the Service provisioning messages of [BCAST10-Services] to 'PurchaseURL' to enable the service provisioning functionality. 2: The Terminal SHALL be able to access the 'PortalURL' via HTTP to acquire further information on purchase items available on this purchase channel. Further, the Terminal MAY perform the service provisioning directly on the 'PortalURL' or send the Service provisioning messages of [BCAST10-Services] to 'PurchaseURL' to enable the service provisioning functionality. 3-255: Reserved for future use. Default: 0	unsignedByte
PurchaseUR	E1	NM/	0..N	The URL to which the BCAST Service	anyURI

L		TM		Provisioning messages as specified in section 5.1 of [BCAST10-Services] SHALL be addressed.	
Extension	E1	NM/ TM	0..N	Additional information related to this fragment. Contains the following attribute: url Contains the following element: Description	
url	A	NM/ TM	1	URL containing additional information related to this fragment.	anyURI
Description	E2	NM/TM	0..N	Description regarding the additional information which can be retrieved from a web page. The language is expressed using built-in XML attribute 'xml:lang' with this element	string
				End of program guide	
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.9 PreviewData

'PreviewData' fragment contains information about the preview data that is used by the terminal to present the service or content outline to users, so that the users can have a general idea of what the service is about. It can also be used to present an alternative to a blacked-out service. The 'PreviewData' fragment can include simple texts, static binary images (for example, logo), references to images, short video clips and audio clips, or even reference to another service which could be a low bit rate version for the main service. Other fragments like 'Service', 'Content', 'Schedule', 'PurchaseData' and 'Access' can link to 'PreviewData' fragment.

Typically the preview data can be used for service-by-service switching, Service Guide browsing, service preview, barker, carrying representation of Terms of Use, alternative in case of blackout, etc. How to use the preview data in different circumstances are further specified in section 5.7.

The network MAY deliver the preview data over broadcast channel using BCAST File Distribution functionality or BCAST Stream Distribution Functionality. Alternatively, the network MAY deliver the preview data over interaction channel.

The terminal SHALL support reception of preview data over broadcast channel. The distribution session for preview data distribution is specified by the 'Access' fragment which is referenced by AccessReference within 'PreviewData' fragment.

- If the 'Usage' attribute of 'AccessReference' element has the value of "0"; the media components of the preview data are delivered via a file distribution session and are identified by URIs, i.e.VideoURI, Audio URI, PictureURI. The distribution scheduling is signaled by Session Description embedded in or referenced by the 'Access' fragment. In this case, the preview data SHALL be distributed by the Network before time of distribution of the associated main service, so that the terminal can retrieve, store and show the preview data before the main service is available.
- If the 'Usage' attribute of 'AccessReference' element has the value of "1"; the preview data are delivered via a stream distribution session as a continuous stream which can be a simplified version of the original service. The distribution scheduling is signaled by Session Description embedded in or referenced by the 'Access' fragment.

If the terminal supports interactive channel, the terminal SHALL support the reception of preview data over interaction channel.

Name	Type	Category	Cardinality	Description	Data Type
PreviewData	E			'PreviewData' fragment. Contains the following attributes: id version validFrom validTo Contains the following elements: SMIL Video Audio Picture Text AccessReference PrivateExt	
id	A	NM/ TM	1	ID of the 'PreviewData' fragment. The the value of this attribute SHALL be globally unique. This ID can be used by other fragments so as to link different service descriptions, purchase information, etc. to the PreviewData.	anyURI
version	A	NM/ TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no 'validFrom' attribute is given.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
SMIL	E1	NM/ TM	0..1	Embedded SMIL to define and synchronize the multimedia components (video, audio, picture, text) of the preview data. SMIL (Synchronized Multimedia Integration Language) is a language that allows authors to be able to easily define and synchronize multimedia elements (video, sound, still images) for Web-like presentation and interaction. Contains the following attribute: type	string
type	A	NM/ TM	0..1	The type of SMIL profile associated with this PreviewData. 0: 3GPP PSS SMIL Profile [3GPP TS 26.246] 1: 3GPP2 MSS SMIL Profile [3GPP2 C.S0050] 2 - 127: reserved for future use 128 - 255: reserved for proprietary use If this attribute is not present, the SMIL profile is	unsignedByte

				unspecified.	
Video	E1	NM/ TM	0..1	Video defines how to obtain an audio/video trailer clip which can enable the user to preview the service or content. Contains the following elements: VideoURI MIMEType AlternativeText AlternativePicture	
VideoURI	E2	NM/ TM	1	The URI referencing the video clip.	anyURI
MIMEType	E2	NM/ TM	0..1	MIME Media type of the video clip. Contains the following attribute: codec	string
codec	A	NO/ TM	0..1	The codec parameters for the associated MIME Media type. If the file's MIME type definition specifies mandatory parameters, these MUST be included in this string. Optional parameters containing information that can be used to determine as to whether the terminal can make use of the file SHOULD be included in the string. One example of the parameters defined for video/3GPP, video/3GPP2 is specified in [RFC4281].	string
AlternativeText	E2	NM/ TM	0..N	Alternative Text to be displayed if the video clip is not available. Possibly in multiple languages. The language is expressed using built-in XML attribute ‘xml:lang’ with this element. The same schema of element ’Text‘ is used for ’AlternativeText’.	
AlternativePicture	E2	NO/ TM	0..1	Alternative Picture to be displayed if the video clip is not available. AlternativePicture can be PictureData or URI reference of the Picture. The same schema of element ‘Picture’ is used for ‘AlternativePicture’	
Audio	E1	NM/ TM	0..1	Audio defines how to obtain an audio clip which can enable the user to preview the service or content. Contains the following elements: AudioURI MIMEType AlternativeText AlternativePicture	
AudioURI	E2	NM/ TM	1	The URI referencing the audio clip.	anyURI
MIMEType	E2	NM/ TM	0..1	MIME Media type of the audio clip. Contains the following attribute: codec	string
codec	A	NO/	0..1	The codec parameters for the associated MIME	string

		TM		Media type. If the file's MIME type definition specifies mandatory parameters, these MUST be included in this string. Optional parameters containing information that can be used to determine as to whether the terminal can make use of the file SHOULD be included in the string. One example of the parameters defined for audio/3GPP, audio/3GPP2 is specified in [RFC4281].	
AlternativeText	E2	NM/ TM	0..N	Alternative Text to be displayed if the audio clip is not available. Possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element. The same schema of element 'Text' is used for 'AlternativeText'.	
AlternativePicture	E2	NO/ TM	0..1	Alternative Picture to be displayed if the audio clip is not available. AlternativePicture can be PictureData or URI reference of the Picture. The same schema of element 'Picture' is used for 'AlternativePicture'.	
Picture	E1	NM/ TM	0..1	Picture defines how to obtain a picture which can enable the user to preview the service or content, or represents the service or content with an icon, a logo, etc. The associated picture can be represented with binary data embedded in the 'PreviewData' fragment or be referenced by URI Contains the following elements: PictureURI PictureData MIMEType AlternativeText	
PictureURI	E2	NM/ TM	0..1	The URI referencing the picture. Either PictureURI or PictureData MUST be used if Picture element is present.	anyURI
PictureData	E2	NM/ TM	0..1	The binary data of the picture, encoded in Base64 to allow embedding into XML. Either PictureURI or PictureData MUST be used if Picture element is present.	base64Binary
MIMEType	E2	NM/ TM	0..1	MIME Media type of the picture.	string
AlternativeText	E2	NM/ TM	0..N	Alternative Text to be displayed if the picture is not available. Possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element. The same schema of element 'Text' is used for 'AlternativeText'.	
Text	E1	NM/ TM	0..N	Text represents textual preview of the service or content. For example, the contents of this element can be a title sentence, or brief description of service or content for preview purposes. Possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element.	string

				Text format attributes (e.g. font, size and colour) are defined by HTML version 4.01. This element SHALL NOT override the value of element 'Name' in 'Service' or 'Content' fragment.	
AccessReference	E1	NM/ TM	0..1	ID of the 'Access' fragment that specifies the delivery of the associated preview data via broadcast channel, during a file distribution session or stream distribution session Contains the following attributes: idRef usage	
idRef	A	NM/ TM	1	Identification of the 'Access' fragment which this 'PreviewData' fragment is associated with.	anyURI
usage	A	NM/ TM	1	Usage of the 'Access' fragment identified by 'AccessReference' element. Allowed value: 0. Indicates that the preview data files referenced by VideoURI, AudioURI or PictureURI should be accessed from a file distribution session, whose distribution scheduling is signaled by Session Description embedded in or referenced by the 'Access' fragment 1. Indicates that the preview data stream should be accessed from a stream distribution session, whose distribution scheduling is signaled by Session Description embedded in or referenced by the 'Access' fragment. 2-127. reserved for future use 128 -255. reserved for proprietary use	unsignedByte
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/ TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.1.2.10 InteractivityData

The 'InteractivityData' fragment is used to associate services and/or individual pieces of content of the services with interactivity components of service/content consumption. These interactivity components are used by the terminal to offer interactive services to the user possibly in parallel with the 'regular' broadcast content. These interactivity services enable users to e.g. vote during TV shows or to obtain content related to the 'regular' broadcast content. Whereas the 'InteractivityData' fragment can be thought to declare the availability of the interactivity components, the details of the components are provided via one or many InteractivityMediaDocuments (see [BCAST10-Services] section 5.3.6.1) that may include xhtml files, static images, email template, SMS template, MMS template documents, etc.

The 'InteractivityData' fragment SHALL always refer to one 'Service' fragment. The 'InteractivityData' fragment can also reference 'Content' fragment, 'Schedule' fragment or Interactivity Windows to further narrow down the association by linking the interactivity with part of the service. The presence of 'ContentReference' element, 'ScheduleReference' element and 'InteractivityWindow' element in the 'InteractivityData' fragment SHALL be mutually exclusive.

Whenever there are more than one 'InteractivityData' fragments valid at the same time for a certain service, the following priority SHALL be maintained by the terminal for processing and rendering the associated interactivity. However how the terminal actually processes and renders the simultaneous interactivities with different priorities is out of the scope of this specification.

- Priority 1 (Highest) – 'ServiceReference' element and 'InteractivityWindow' element are present in the InteractivityData' fragment:

  In this case the interactivity is associated with the declared interactivity windows of this service and the interactivity can be activated during those interactivity windows. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments.

- Priority 2 – 'ServiceReference' element, 'ScheduleReference' element and 'ScheduleReference.PresentationWindowIDRef' sub-element are present:

  In this case the interactivity is associated with the indicated subset of the presentation windows of the service which are originally declared in the referenced Schedule fragment, and this interactivity can be activated during these sub-set of presentation windows. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments.

- Priority 3 – 'ServiceReference' element and 'ScheduleReference' element are present, but 'ScheduleReference.PresentationWindowIDRef' sub-element is not present:

  In this case the interactivity is associated with all presentation windows of this service which are originally declared in the referenced Schedule fragment, and this interactivity can be activated during all these presentation windows. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments.

- Priority 4 – 'ServiceReference' element and 'ContentReference' element are present

  In this case the interactivity is associated with the content of this service and can be activated during the life span of the content. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments.

- Priority 5 (Lowest) – only 'ServiceReference' element is present

  In this case the interactivity is associated with this service and can be activated during the life span of the service. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments.

Terminals with no return channel MAY NOT support 'InteractivityData' fragments. Terminals with a return channel (e.g. UMTS, GPRS) SHALL support 'InteractivityData' fragments.

The concept of grouping of InteractivityMediaDocuments (as defined in [BCAST10-Services] section 5.3.6.1) allows

- to group multiple InteractivityMediaDocuments into one logical unit representing the same interactivity,
- but also to associate InteractivityMediaDocuments with 'InteractivityData' fragments.

The network SHALL assign each 'InteractivityData' fragment one and only one InteractivityMediaDocument group identifier (InteractivityMediaDocumentPointer) which is used by the terminals to identify the InteractivityMediaDocument belonging to the interactivity represented by the 'InteractivityData' fragment in question.

The media objects listed in the InteractivityMediaDocuments associated with the 'InteractivityData' fragment can be distributed before or at the same time of distribution of the 'regular' broadcast media stream. An InteractivityMediaDocument distributed before that time can be cached by the terminal. This is indicated by setting the Pre-listenIndicator to "true".

InteractivityMediaDocuments can be distributed over the same access channel as the service they are associated with, or over a different access channel. Distribution over a different access is enabled by association of an 'InteractivityData' fragment to a 'Schedule' fragment that is referred to by a different 'Access' fragment than service.

In case multiple groups of InteractivityMediaDocuments are transported using the same file delivery session, the network MUST assign mutually exclusive InteractivityMediaDocument group identifiers to the 'InteractivityData' fragments in question. This is in order to enable the terminals to distinguish between the InteractivityMediaDocuments of the different interactivities.

'InteractivityData' fragment can specify that interaction sent back from device to service provider shall be distributed over time, e.g. to avoid overload in network nodes or links caused by too many simultaneous interactivity messages sent back. This is done by declaring a time window during which interactivity shall be sent back. The declaration of this information in an 'InteractivityData' fragment applies to all interactivity declared through this 'InteractivityData' fragment, but can be superseded by explicit declaration of the same information per interaction, in an InteractivityMediaDocument.

If interaction is announced using 'InteractivityData' fragment, the service provider SHOULD deliver InteractivityMediaDocuments corresponding to that 'InteractivityData' fragment.

Name	Type	Category	Cardinality	Description	Data Type
Interactivity Data	E			'InteractivityData' fragment. Contains the following attributes: id version validFrom validTo preListenIndicator interactivityMediaDocumentPointer Contains the following elements: InteractivityType ServiceReference ContentReference ScheduleReference InteractivityWindow InteractiveDelivery Extension BackOffTiming TermsOfUse PrivateExt	
id	A	NM/TM	1	ID of the 'InteractivityData' fragment. The the value of this attribute SHALL be globally unique.	anyURI
version	A	NM/TM	1	Version of this fragment. The newer version overrides the older one starting from the time specified by the 'validFrom' attribute, or as soon as it has been received if no 'validFrom' attribute is given.	unsignedInt
validFrom	A	NM/TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp	unsignedInt
validTo	A	NM/TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp	unsignedInt
preListenIndicator	A	NM/TM	1	If the attribute 'prelistenIndicator' is "true" the terminal SHOULD retrieve and locally store the Interactivity media objects included in the InteractivityMediaDocuments carried in the	boolean

				broadcast stream (see [BCAST10-Services] section 5.3.6). The terminal SHOULD start the retrieval of these Interactivity media objects prior to the broadcast time of the 'Service', 'Content', 'Schedule' or 'InteractivityWindow' it is associated with or as soon as the 'InteractivityData' fragment is retrieved by the terminal. If the attribute 'prelistenIndicator' is "false" the terminal MAY retrieve the Interactivity media objects included in the InteractivityMediaDocuments, before the 'Service', 'Content', 'Schedule' or 'InteractivityWindow' it is associated with, is broadcasted.	
interactivityMediaDocumentPointer	A	NM/TM	1	Reference to the GroupID of the InteractivityMediaDocuments which refer to the interactivity media objects. The pointer points to all InteractivityMediaDocuments with the same GroupID. The InteractivityMediaDocument with the highest GroupPosition (see [BCAST10-Services] section 5.3.6) is rendered. When multiple 'InteractivityData' fragments point to the same GroupID, this means that the InteractivityMediaDocuments belonging to this GroupID applies to all of these 'InteractivityData' fragments.	anyURI
InteractivityType	E1	NO/TO	0..N	Description of the type of the Interactive Service, such as "voting" or "gambling", possibly in multiple languages. The language is expressed using built-in XML attribute 'xml:lang' with this element. Terminal devices supporting Interactive channel SHALL support this attribute. It SHALL be used for rendering only.	string
ServiceReference	E1	NM/TM	1	Reference to the 'Service' fragment that the 'InteractivityData' fragment is associated with. Contains the following attribute: idRef	
idRef	A	NM/TM	1	Identification of the 'Service' fragment which this 'InteractivityData' fragment is associated with.	anyURI
ContentReference	E1	NM/TM	0..N	Reference to the 'Content' fragments that the 'InteractivityData' fragment is associated with. If this element is present, that means the interactivity is associated with the referenced content and may be activated during the life span of the content. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments. The presence of 'ContentReference' element,	

				'ScheduleReference' element and 'InteractivityWindow' element SHALL be mutually exclusive. Implementation in XML Schema is done using <choice>. Contains the following attribute: idRef	
idRef	A	NM/TM	1	Identification of the 'Content' fragment which this 'InteractivityData' fragment is associated with.	anyURI
ScheduleReference	E1	NM/TM	0..N	Reference to the 'Schedule' fragments that the 'InteractivityData' fragment is associated with. If this element is present, that means the interactivity is associated with the referenced schedule, and may be activated during all Presentation Windows or a subset of the presentation windows indicated by 'PresentationWindowIDRef' sub-element. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments. The presence of 'ContentReference' element, 'ScheduleReference' element and 'InteractivityWindow' element SHALL be mutually exclusive. Implementation in XML Schema is done using <choice>. Contains the following attribute: idRef Contains the following element: PresentationWindowIDRef	
idRef	A	NM/TM	1	The id of the 'Schedule' fragment this 'InteractivityData' fragment refers to, globally unique.	anyURI
PresentationWindowIDRef	E2	NM/TM	0..N	Relation reference to the PresentationWindow to which the 'InteractivityData' fragment belongs. The 'PresentationWindowIDRef' declared in this sub-element SHALL be the complete collection or a subset of the PresentationWindow ids declared in the 'Schedule' fragment referenced by the above Schedule 'idRef'. If the latter case is true, then the interactivity SHALL only be activated during these presentation windows, as opposed to during every presentation windows declared in the referenced 'Schedule' fragment. If 'PresentationWindowIDRef' element is absent, the interactivity SHALL be activated during every presentation windows declared in the referenced 'Schedule' fragment.	unsignedInt
InteractivityWindow	E1	NM/TM	0..N	Time interval during which this 'InteractivityData' fragment is associated with the	

				service specified by 'ServiceReference' element. If this element is present, that means the interactivity is associated with the declared InteractivityWindows and may be activated during those InteractivityWindows. The actual validity for the terminal to access the interactivity is declared by 'validFrom' and 'validTo' attributes in the corresponding InteractivityMediaDocuments. The presence of 'ContentReference' element, 'ScheduleReference' element and 'InteractivityWindow' element SHALL be mutually exclusive. Implementation in XML Schema is done using <choice>. Contains the following attributes : startTime endTime	
startTime	A	NM/TM	1	Start of the InteractivityWindow. Whenever an InteractivityWindow is specified, StartTime SHALL be declared. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
endTime	A	NM/TM	1	End of the InteractivityWindow. Whenever an InteractivityWindow is specified, EndTime SHALL be declared. This field contains the 32bits integer part of an NTP time stamp.	unsignedInt
InteractiveDelivery	E1	NO/TM	0..1	This element indicates the possibility to receive InteractivityMedia over the interaction channel. Interactivity Media can either be pushed, using OMA PUSH delivery, or pulled, using HTTP requests to InteractivityMediaURL. If this element is present, at least one of PushDelivery and InteractivityMediaURL shall be included. Contains the following attributes: interactivtyMediaURL pushDelivery	
interactivtyMediaURL	A	NO/TM	0..1	URL from which Interactivity Media can be retrieved. The Content-Type SHALL be "multipart/mixed" in the HTTP response.	anyURI
pushDelivery	A	NO/TM	0..1	If this attribute is present and "true", the terminal SHALL expect the delivery of InteractivityMedia using OMA PUSH, as described in [BCAST10-Distribution] section 9. In this case, the PUSH messages contain InteractivityMedia whose content type is "application/vnd.oma.bcast.imd+xml". The default of this attribute is "false".	boolean
Extension	E1	NM/TM	0..1	Additional information related to this fragment. Contains the following attribute: url	

				Contains the following element: Description	
url	A	NM/ TM	1	URL containing additional information related to this fragment.	anyURI
Description	E2	NM/TM	0..N	Description regarding the additional information which can be retrieved from a web page. The language is expressed using built-in XML attribute xml:lang with this element	string
BackOffTiming	E1	NM/TM	0..1	This element specifies default timing behaviour of interaction sent back from the device to the service provider. Its purpose is to provide a mechanisms that ensures distribution over time of feedback sent from receivers, e.g. in order to avoid overload in nodes or links. If present, the interaction, if any, SHALL be sent back in the time interval [offsetTime, offsetTime+randomTime] after the event that triggers the interactivity (e.g. user feedback). The exact time within the allowed time window shall be random with uniform probability. The declaration of this information in InteractivityData applies as a default to all interactivity declared through this InteractivityData, but if explicit timing behaviour is expressed in InteractivityMediaDocument, it prevails over default timing behaviour expressed in 'InteractivityData' fragment. Contains the following attributes: offsetTime randomTime	
offsetTime	A	NM/TM	1	The 'offsetTime' attribute specifies the minimum time that a device SHALL wait after an event that triggers interaction (e.g. user input), before sending the interaction. The unit is seconds (fractions can be expressed using data type Decimal). 'offsetTime' shall be a non-negative number.	decimal
randomTime	A	NM/TM	1	The 'randomTime' refers to the time window length over which a device SHALL calculate a random time for the transmission of interaction. The method provides for statistically uniform distribution over a relevant period of time. The device SHALL calculate a uniformly distributed random time out of the interval between 0 and randomTime. The unit is seconds (fractions can be expressed using data type Decimal). 'randomTime' shall be a non-negative number.	decimal
TermsOfUse	E1	NO/TO	0..N	Element that declares there are Terms of Use associated with this fragment. Contains the textual presentation of Terms of Use or a reference to Terms of Use representation through 'PreviewData', and information whether	

				user consent is required for the Terms of Use. Multiple occurrences of 'TermsOfUse' are allowed within this fragment, but for any two such occurrences values for elements 'Country' and 'Language' SHALL NOT be same at the same time. In addition to Terms of Use this element MAY be used for disclaimers, legal text and other pieces of information to be rendered to the user upon activation, purchase or consumption of service or content. Contains the following attributes: type id userConsentRequired Contains the following elements: Country Language PreviewDataIDRef TermsOfUseText	
type	A	NM/TM	1	The way the terminal SHALL interpret the Terms of Use: 0 – Display before purchasing or subscribing. If 'TermsOfUse' element of type '1' is present, terminal SHALL render the Terms of Use prior to initiating purchase or subscription request related PurchaseItem associated with this fragment. 1 – Display before playout. If 'TermsOfUse' element of type '2' is present, terminal SHALL present the Terms of Use prior to playing out content or service associated with this fragment. 2 - 127 reserved for future use 128 -255 reserved for proprietary use	unsignedByte
id	A	NM/TM	1	The URI uniquely identifying the Terms of Use.	anyURI
userConsent Required	A	NM/TM	1	Signals whether user consent for these Terms of Use is needed. true: User consent is required for these Terms of Use and needs to be confirmed in the subscription / purchase request message related to the PurchaseItem associated with this fragment. false: User consent is not required for the Terms of Use.	boolean
Country	E2	NM/TM	1..N	List of countries for which the Terms of Use is applicable. Each value is a three character string according to ISO 3166-1 alpha-3	string
Language	E2	NM/TM	1	Language in which the Terms of Use is given. Value is a three character string according to ISO	string

				639-2 alpha standard for language codes.	
PreviewData IDRef	E2	NO/TM	0..N	Reference to the 'PreviewData' fragment which carries the representation of Terms of Use. If this element is not present, the 'TermsOfUseText' SHALL be present.	anyURI
TermsOfUse Text	E2	NO/TM	0..1	Terms of Use text to be rendered. If 'PreviewDataIDRef' element is present under the 'TermsOfUse' this element SHALL NOT be present.	string
PrivateExt	E1	NO/TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.2 Announcing Service Guides within a Service Guide

It is possible to announce other Service Guides within a Service Guide. For the sake of clarity, the Service Guide that refers to another Service Guide is called *referring Service Guide* while the Service Guide that is referred to is called *referred Service Guide*. There are two use cases that announcing other Service Guides within a Service Guide enables:

- **Service Guide complementing another Service Guide**: For example, basic (referring) Service Guide may describe basic services on top level and while the complementing (referred) Service Guides may describe further details of services and/or additional services. In this case the referred Service Guides will need the baseline information in the referring Service Guide.
- **Referring to standalone Service Guide**: For example, basic (referring) Service Guide may aggregate a number of stand-alone (referred) Service Guides by reference. In this case the referred Service Guides will be fully usable even without the referring Service Guide.

The following applies when Service Guides are announced from other Service Guides:

- If the existence of and access to referred Service Guide is announced, the 'Access' fragment SHALL be used to announce those:
    - Referring Service Guide MAY include 'ServiceClass' element in the 'Access' fragment with value urn:oma:bcast:oma_bsc:csg:1.0 meaning that the referred Service Guide provides complementary information to the referring Service Guide.
    - Referring Service Guide MAY include 'ServiceClass' element in the 'Access' fragment with value urn:oma:bcast:oma_bsc:sg:1.0 meaning that the referred Service Guide is a stand-alone Service Guide.
- If all 'Access' fragments associated with a 'Service' fragment announce referred Service Guides, the value of the 'ServiceType' element of 'Service' fragment SHALL be set to "8 – Service Guide".
- If the referred Service Guide is complementary Service Guide, the SG-C SHOULD assume the existence of referring Service Guide to make use of information provided in the referred Service Guide.
- If the referred Service Guide is stand-alone Service Guide, the SG-C SHOULD NOT assume the existence of referring Service Guide to make use of information provided in the referred Service Guide.

SG-C SHALL support announcing other Service Guides within a Service Guide and SG-D MAY announce other Service Guides within a Service Guide as defined above.

5.3 Interfaces for Service Guide Generation

Referring to OMA BCAST Architecture [BCAST10-Architecture], this section normatively specifies the interfaces SG-1, SG-2, SG-4, SG-5 and SG-6 for Service Guide functionality.

5.3.1 Service Guide Data Model as Generic Interchange format

The Service Guide Data Model (as specified in section 5.1) and its XML Schema definition (as specified in [BCAST10-Schema-sg-f]) SHALL be used as generic interchange format for communicating Service Guide entries as fragments over interfaces SG-2, SG-4 and SG-B1 (system back-end interfaces) and over interfaces SG-5 and SG-6 (interfaces between terminal and system). Over interface SG-1 the BCAST XML schema definition MAY be used.

The methods to deliver Service Guide fragments over interfaces SG-1, SG-2 and SG-4 are defined in section 5.4.

The methods to deliver Service Guide fragments over interface SG-B1 are specific to each BDS in question and are covered in the BDS adaptation specifications (see [BCAST10-DVBH-IPDC-Adaptation], [BCAST10-MBMS-Adaptation], [BCAST10-BCMCS-Adaptation]).

The method to deliver Service Guide fragments over interface SG-5 is specified in section 5.4.2.

The method to deliver Service Guide fragments over interface SG-6 is specified in section 5.4.3.

5.4 Service Guide Delivery

Instantiated Service Guide consists of the Service Guide XML fragments and the Session Description information fragments described in the previous sections. Each fragment is designed to be uniquelly identifiable in the Service Guide making it possible to treate the Service Guide as a set of these fragments. This naturally makes it possible for the network to divide the set further into subsets each containing only some of the fragments in the set but not necessarily all of them. The network can deliver the Service Guide using these subsets each subset independent of the other instead of placing the full set to the transport as one atomic unit. This arrangement enables the network to isolate the changes on the delivery layer when only some of the fragments in the Service Guide are altered. In such a case the terminals that have already received the full set of fragments at least once, can utilise the isolation of the changes by receiving only that subset of fragments that contains the changes.

The delivery of the Service Guide can be provided by two means

- over the interactive channel the terminal requesting some or all of the fragments of the Service Guide explicitly or
- over the broadcast channel the terminals polling the fragments.

While both mechanisms have their unique characteristics the mechanisms essentially share the data structures used to manage the organization of the fragments on the transport layer. These data structures are the ‘Service Guide Delivery Descriptor’ and the ‘Service Guide Delivery Unit’ short-handed SGDD and SGDU respectively. The latter is the structure that the network uses to encapsulate fragment subsets for the transport layer and it is purely transport independent. The former, the SGDD, on the other hand is partially transport dependent. The SGDD both describes service level information about the Service Guide as well how each of the Service Guide fragment is available for the terminal in the transport layer.

The following sections are organised to address transport layer independent mechanisms and definitions first followed by the transport level dependent sections.

5.4.1 Encapsulating and identifying fragments

As discussed before, each of the Service Guide fragments as an XML fragment or Session Description is assigned an identifier for distinguishing any two fragments from each other. These identifiers are called fragment identifiers and they are represented with URIs. This identification scheme naturally makes the fragments not only unique in the scope of a single Service Guide but in fact over all the Service Guides.

In the transport frames, however, one prefers the use of integer based identification schemes over string based identification schemes. This is achieved by assigning for each fragment in addition to the fragment identifier a transport identifier. In the

transport layer the fragments are addressed using the transport identifiers the transport layer also providing the binding between these two types of identifiers.

While the specifics of the common data structures SGDD and SGDU are discussed in the later sections, the strong dependency between the two structures imposed by the transport identifiers is defined first.

5.4.1.1 Fragment identifier versus transport identifier

As discussed before any two Service Guide fragments can be distinguished by their fragment identifiers. The way these identifiers are represented depends on the type of the fragment:

- In case of XML fragments the identifier can be found in the fragment itself as top level attribute 'id' and
- In case of Session Description the identifier is introduced in the SGDU carrying the fragment using the field 'fragmentID'.

The transport identifiers used in the transport layer to identify fragments are given in the 'fragmentTransportID' fields of the SGDUs carrying the corresponding fragments. The binding between the fragment identifier and the transport identifier is provided using the SGDD where each fragment is declared using an XML element 'Fragment'. This element gives both the fragment identifier and the transport identifier for the fragment providing explicit binding between the two. In order for the binding to be unambiguous the network SHALL ensure the following:

- During the whole lifetime of a Service Guide fragment the 'id' attribute of the fragment is always bound to the same 'fragmentTransportID' value.
- For each Service Guide entry point (see section 6.1.1), the binding between 'fragmentTransportID' and 'id' attribute is one-to-one.

The terminal SHALL comply with the following rules on maintaining the binding between fragment identifiers and transport identifiers in the terminal:

- If a terminal has received a Service Guide Delivery Descriptor (SGDD), the terminal SHALL record the binding between the 'fragmentTransportID' fields and the corresponding 'fragmentID' fields of the 'Fragment' element in that descriptor.
- If a terminal has not encountered the binding for a particular 'fragmentTransportID' in the SGDD, the terminal SHALL decode the corresponding fragment delivered in the SGDU, associate the decoded fragment identifier with the 'fragmentTransportID' in the SGDU header and record that binding.
- A terminal SHALL NOT use any recorded 'fragmentID'-to-'fragmentTransportID' binding after the fragment expires.

5.4.1.2 Resolving fragment references

Some Service Guide fragments have a capability to refer to other Service Guide fragments. This is achieved by including a reference element to the referring fragment. The element is named '<*fragment name*>Reference' and the value of its attribute 'idRef' is of type 'anyURI' – this value contains the identifier of the referred fragment. The terminal can obtain the referenced fragment in two ways either receiving it over broadcast channel or retrieving it over interactive channel.

- When receiving the referred fragment over broadcast channel the 'id' attribute of the referred fragment is resolved to 'transportID' and consequently to 'fragmentTransportID' within the SGDU as specified in section 5.4.1.
- When retrieving the referred fragment over interactive channel, the delivery method as specified in section 5.4.3 (the case when a fragment is requested using the individual fragment ID) is used. In this context, the referred fragment id is used as a value in a name-value pair of attribute 'fragmentID' in the outgoing interactive request message.

All the terminals SHALL support receiving the referred fragment(s) over broadcast channel, while all terminals able to access the interaction channel SHALL support requesting and receiving the referred fragment(s) over the interaction channel also.

5.4.1.3 Service Guide fragment encapsulation

In order to deliver the fragments from the network to the terminals the network needs to be able to place the fragments into the underlying transport frames. The network is provided with means of delivering more than one fragment as a atomic unit at the time but on the other hand the network is not restricted to deliver al the fragments at one go either. For the terminals to correctly receive and process any collection of fragments as one delivery unit the network SHALL comply with the following:

- The Service Guide Delivery Unit structure as defined in Table X1 SHALL be used for encapsulating Service Guide fragments for transport.
- The field 'fragmentTransportID' SHALL be assigned with the 'transportID' values as defined in section 5.4.1.1 to identify each of the fragments carried in the Service Guide Delivery Unit.
- When encapsulating the fragments into the Service Guide Delivery Unit, the mapping defined in section 5.4.1.1 SHALL be used.
- In case the SGDUs are listed in any FDT Instances the corresponding 'Content-Type' attributes SHALL be set to "application/vnd.oma.bcast.sgdu" to describe that the transport object contains an SGDU.

Using the 'fragmentTransportID' and 'fragmentVersion' fields the terminal can quickly infer whether the associated fragment in the SGDU has changed.

Data Field Name	Data Type
Service_Guide_Delivery_Unit {	
Unit_Header {	
extension_offset	uimsbf32
Reserved	16 bits
n_o_service_guide_fragments	uimsbf24
for(i=0; i< n_o_service_guide_fragments; i++) {	
fragmentTransportID[i]	uimsbf32
fragmentVersion[i]	uimsbf32
offset[i]	uimsbf32
}	
}	
Unit_Payload {	
for(i=0; i< n_o_service_guide_fragments; i++) {	
fragmentEncoding[i]	uimsbf8
if(fragmentEncoding[i]=0) {	
fragmentType	uimsbf8
XMLFragment	bitstring
}	
else if(fragmentEncoding[i]=1) {	
validFrom	uimsbf32
validTo	uimsbf32
fragmentID	bitstring
SDPfragment	bitstring
}	
else if(fragmentEncoding[i]=2) {	
validFrom	uimsbf32
validTo	uimsbf32
fragmentID	bitstring
USBDfragment	bitstring

}	
else if(fragmentEncoding[i]=3) {	
validFrom	uimsbf32
validTo	uimsbf32
fragmentID	bitstring
ADPfragment	bitstring
}	
}	
}	
if(extension_offset>0) {	
extension_type	uimsbf8
next_extension_offset	uimsbf32
Reserved	bitstring
}	
}	

Table 1: Service Guide Delivery Unit structure

uimsbfN	Unsigned Nbit Integer, most significant bit first
bitstring	Array of bits

Table 2: Mnemonics used in Table 1

extension_offset	Offset in bytes from the start of the Unit_Payload to the start of the first extension. Set to 0 if there is no extension present.
n_o_service_guide_fragments	Number of Service Guide fragments encapsulated in this specific Delivery Unit.
offset[i]	Offset in bytes from the start of the Unit_Payload to the start of the fragment_encoding field of the i:th Service Guide fragment. The offset list is sorted in ascending order.
fragmentTransportID[i]	Signals the identifier of the i:th Service Guide fragment which is defined for transport (see 5.4.1.5)
fragmentVersion[i]	Signals the version of the i:th Service Guide fragment. Note: The scope of the fragmentVersion is limited to this transport session. The value of fragmentVersion can turn over from 2^32-1 to 0.
fragmentEncoding[i]	Signals the encoding of the i:th Service Guide fragment, with the following values: 0 – XML encoded OMA BCAST Service Guide fragment 1 – SDP fragment 2 – MBMS User Service Bundle Description (USBD) as specified in [26.346] (see 5.1.2.4 'SessionDescription' element) 3 – XML encoded Associated Delivery Procedure as specified in [BCAST10-Distribution] section 5.3.4. 4-127 – reserved for future BCAST extensions 128-255 – available for proprietary extensions
fragmentType[i]	This field signals the type of an XML encoded BCAST Service Guide fragment, with the following values: 0 – unspecified

	1 – 'Service' Fragment 2 – 'Content' fragment 3 – 'Schedule' Fragment 4 – 'Access' Fragment 5 – 'PurchaseItem' Fragment 6 – 'PurchaseData' Fragment 7– 'PurchaseChannel' Fragment 8 – 'PreviewData' Fragment 9 – 'InteractivityData' Fragment 10-127 – reserved for BCAST extensions 128-255 – available for proprietary extensions
fragmentID	Null-terminated string containing the fragment ID of an SDP or MBMS USBD or Associated Delivery Procedure fragment as referenced from an 'Access' fragment via SessionDescriptionReference. Note: for an XML encoded OMA BCAST Service Guide fragment, this information is contained in the fragment itself.
validFrom	32 bit word representation of the validFrom value of an SDP or MBMS USBD or Associated Delivery Procedure fragment. This field is expressed as the first 32bits integer part of NTP time stamp. When set to "0" the interpretation is that "validFrom" is undefined. Note: for an XML encoded OMA BCAST Service Guide fragment, this information is contained in the fragment itself.
validTo	32 bit word representation of the the validTo value of an SDP or MBMS USBD or Associated Delivery Procedure Description fragment. This field is expressed as the first 32bits integer part of NTP time stamp. When set to "0" the interpretation is that "validTo" is undefined. Note: for an XML encoded OMA BCAST Service Guide fragment, this information is contained in the fragment itself.
XMLfragment	String containing the actual XML data of the encapsulated Service Guide fragment without the termination character.
SDPfragment	String containing the actual SDP data, without termination character.
USBDfragment	String containing the actual MBMS USBD data, without termination character.
ADPfragment	String containing the actual XML data of the encapsulated Associated Delivery Procedure fragment, without termination character.
extension_type	Signals the type of the extension. 0-127 – reserved for BCAST extensions 128-255 – available for proprietary extensions
next_extension_offset	Offset in bytes from the start of the current extension to the start of the next extension. Set to 0 if there is no next extension.

Table 3: Semantics for Table 1

5.4.1.4 Compression of Service Guide Delivery Units

The network is provided with means of reducing the size of the SGDUs being delivered to terminals by compressing SGDUs. For the algorithms and their respective signalling there are the following rules and constraints:

- the network MAY compress the SGDUs with the GZIP algorithm,

- terminals SHALL support both plain SGDUs and GZIP compressed SGDUs,
- in the case of broadcast delivery of SGDUs, the network SHALL signal GZIP compression of every SGDU using the 'encoding' attribute of the SGDD and
- in case the SGDUs are listed in any FDT Instances the corresponding 'Content-Encoding' attributes SHALL be set to "gzip" also.
- Additionally, when FLUTE is used for transmission of the SGDD:
- the network MAY compress the SGDDs with the GZIP algorithm. In this case the Content-Encoding attribute in the corresponding description of the FDT SHALL be set to "gzip".

5.4.1.5 Service Guide Delivery Descriptor

Recall that all the fragments of the Service Guide are not necessarily placed into one and single SGDU but rather into multiple SGDUs. Even though the details of the exact delivery mechanisms are discussed in the later section, one needs to take one fundamental difference between the broadcast and interactive delivery into account here. This is the stragegy used to divide the set of Service Guide fragments into transport layer subsets (each carried by its own SGDUs). In the broadcast case the division is fixed at a certain point of time and the division shared by the network and receiving terminals. In the interactive case there is no fixed division, the network simply provides one subset per request the subset contents being defined by the request.

In the broadcast case the division of fragments into separate transport frames (SGDUs) requires signalling of the details of the division, basicly the SGDUs being used, how the SGDUs can be accessed on the transport and a map telling which fragment can be found in which SGDU. This transport layout signalling is provided by the network using the aforementioned SGDDs, Service Guide Delivery Descriptors, the signalling clearly representing transport dependent part of the information in the SGDDs.

Even though the details of the fixed division of the fragments into SGDUs are not applicable on the interactive delivery case, both the terminals performing the interactive or the broadcast retrieval of the Service Guide, however, rely on the SGDDs to provide exhaustive list of the fragments in the Service Guide.

In addition to dividing the Service Guide fragments into subsets for the transport layer, the network can also divide the fragments into subsets on the service layer independent on the transport layer division. This latter type of division of the Service Guide in the application layer is called grouping of the Service Guide and it is also signalled in the SGDDs. This concept, however, depends on application layer constructs and it is discussed in its own section.

When FLUTE is used for transmission of the SGDD, the network MAY compress the SGDDs with the GZIP algorithm. In this case the Content-Encoding attribute in the corresponding description of the FDT SHALL be set to "gzip".

Next section describes in more detail the transport dependent uses of the SGDD.

5.4.1.5.1 Transport dependencies

Similarly as in the case of the network using more than one SGDU frame for delivering the Service Guide fragments, the network can also use multiple SGDDs for declaring the exhaustive list of the fragments in the Service Guide. In such a case it is easy to see that in order for the terminals to be aware of all the Service Guide fragments, the terminals need to be aware of all the SGDDs the network uses. For the broadcast delivery of the SGDDs, the network SHALL therefore place all the SGDDs representing a Service Guide into one and only one delivery session. This session is called the Service Guide Announcement Channel.The network SHALL also make sure that the SGDDs declare all fragments that are delivered over the broadcast channel.

As mentioned before for interactive delivery of the service guide, the main role of the SGDD is to declare all fragments that describe one or more services. The information about division of the fragments into SGDUs in this case is not essential, since all fragments are retrieved interactively and individually for each terminal, and thus a fixed division into SGDUs does not exist. However, the grouping in the service layer can be used to provide information about fragments belonging to the same service. The SGDD MAY declare fragments that are delivered over the interaction channel, but it SHALL at least declare a set of fragments that allow interactive retrieval of the complete SG. For example, the SGDD could declare only 'Service' fragments. The terminal could then interactively retrieve fragments related to specific selected services, using the request mechanism described in section 5.4.3.

5.4.1.5.2 Service Guide Delivery Descriptor

The following structure SHALL be used for declaring availability, metadata and grouping of the fragments of Service Guide, and for delivery of the SG to point to the actual delivery channel and the delivery unit where the declared fragments are to be delivered. The terminal SHALL support the Service Guide Delivery Descriptor syntax as defined by XML Schema in [BCAST10-Schema-sg-sgdd]. For delivery, the Service Guide Delivery Descriptor SHALL be instantiated as an XML document.

The network is provided with means of reducing the size of the SGDDs being delivered to terminals by compressing SGDDs. For the algorithms and their respective signalling there are the following rules and constraints:

- the network MAY compress the SGDDs with the GZIP algorithm,
- terminals SHALL support both plain SGDDs and GZIP compressed SGDDs,

In the case of broadcast delivery of SGDDs, the network SHALL signal GZIP compression of SGDDs by setting the 'content-encoding' attribute of the FLUTE FDT instances listing the SGDDs.

Name	Type	Category	Cardinality	Description	Data Type
ServiceGuideDeliveryDescriptor	E			The Service Guide Delivery Descriptor Contains the following attributes: id version Contains the following elements: NotificationReception BSMList DescriptorEntry	
id	A	NM/TM	0..1	Unique identifier of the SGDD within one specific SG	anyURI
version	A	NM/TM	0..1	Version of SGDD	unsignedInt
Notification Reception	E1	NM/TM	0..1	Reception information for general Notification Messages. In case of delivery over Broadcast channel, IPBroadcastDelivery specifies the address information for receiving Notification message. In case of delivery over Interaction channel, RequestURL specify address information for subscribing notification, PollURL specify address information for polling notification. When the Notification Message resource pointed by this element provides Notification Messages carrying Service Guide update, those SHALL relate to the currently bootstrapped Service Guide. If this element is present, at least one of the atttributes "IPBroadcastDelivery", "RequestURL", or "PollURL" SHALL be present. Contains the following elements: IPBroadcastDelivery RequestURL PollURL	
IPBroadcast	E2	NM/TM	0..1	Provides IP multicast address and port number for	

Delivery				reception of Notification Messages over the broadcast channel. Contains the following attributes: port address	
port	A	NM/TM	1	General Notification Message delivery UDP destination port number; delivery over Broadcast Channel.	unsignedInt
address	A	NM/TM	1	General Notification Message delivery IP multicast address; delivery over Broadcast Channel.	string
RequestURL	E2	NM/TM	0..1	URL through which the terminal can subscribe to general Notification Messages; delivery over Interaction Channel.	anyURI
PollURL	E2	NM/TM	0..1	URL through which the terminal can poll general Notification Messages over Interaction Channel.	anyURI
BSMList	E1	NO/TM	0..1	Declaration of the BSM Selectors which can be used in the GroupingCriteria sections defined below. Contains the following element: BSMSelector	
BSMSelector	E2	NM/ TM	1..N	Specifies the BSM associated with the fragments in this Service Guide Delivery Unit Allows a terminal to determine whether the SGDU's in this SGDD DescriptorEntry – among the SGDU's that are announced in various DescriptorEntries in various SGDD's – is associated with the terminal's affiliated BSM(s). The terminal's affiliated BSM(s) are represented within terminal as Management Objects with identifier '<X>/ BSMFilterCode'. In case the terminal is has with one or more '<X>/bsmFilterCodesBSMFilterCode' entries, for the interpretation of the BSMSelector within the SGDD the following SHALL apply: • If the BSMFilterCode present in this element matches to any of the '<X>/BSMFilterCode' entries within the terminal, the terminal is able to process, render, interpret and handle the fragments without restrictions. • If the BSMFilterCode present in this element does not match to any of the '<X>/BSMFilterCode' entries within the terminal, the terminal can render, interpret and handle the fragments according to RoamingRules associated with this BSMSelector (identified by the attribute 'id'). In case the terminal does not have these RoamingRules the terminal SHALL NOT render the fragments to the user. The terminal MAY acquire the rules by sending a RoamingRuleRequest to address	

				indicated by attribute "RoamingRuleRequestAddress". In case the terminal has no '<X>/BSMFilterCode' entries, for the interpretation of the BSMSelector within the SGDD the following SHALL apply: • The terminal can render, interpret and handle the fragments according to RoamingRules associated with this BSMSelector (identified by the attribute 'id'). In case the terminal does not have these RoamingRules the terminal SHALL NOT render the fragments to the user. The terminal MAY acquire the rules by sending a RoamingRuleRequest to address indicated by attribute "RoamingRuleRequestAddress". Note: RoamingRuleRequest message, RoamingRules and associated roaming methods are specified in [BCAST10-Services]. Contains the following attributes: id roamingRuleRequestAddress Contains the following elements: BSMFilterCode Name RoamingRule	
id	A	NM/TM	1	Identifier of the BSMSelector. This 'id' is unique within the SGDD in which the BSMSlecotr is declared.	anyURI
roamingRuleRequestAddress	A	NO/TM	0..1	Address to which the terminals can send the RoamingRuleRequests to request RoamingRules associated with this BSMSelector (identified with the 'id' attribute).	anyURI
BSMFilterCode	E3	NM/TM	0..1	The code that specifies this BSMSelector. Contains the following attributes: type mobileCountryCode mobileNetworkCode networkSubsetCode networkSubsetCodeRangeStart networkSubsetCodeRangeEnd serviceProviderCode corporateCode nonSmartCardCode	
type	A	NM/TM	1	The type of bsmFilterCode. 1 – BSMCode (Smart Card Code) This is used if the determination is made based on the country and operator code in the (U)SIM/(R-)UIM/CSIM	unsignedByte

				2 – BSMCode (Non Smart Card Code): This is used if the determination is made based on the country and operator code in the terminal Other values are reserved.	
mobileCountryCode	A	NO/ TM	0..1	Mobile Country Code (3 digits) as specified by [3GPP TS 22.022]. Applicable only when "type" == 1	integer
mobileNetworkCode	A	NO/ TM	0..1	Mobile Network Code (2 digits) as specified by [3GPP TS 22.022]. Applicable only when "type" == 1	integer
networkSubsetCode	A	NO/ TM	0..1	Network Subset Code (2 digits) as specified by [3GPP TS 22.022]. Applicable only when "type" == 1	integer
networkSubsetCodeRangeStart	A	NO/ TM	0..1	Instead of providing an explicit code in attribute 'networkSubsetCode', the network MAY instead provide a continuous range of codes. In such a case the network SHALL • provide the smallest code for the terminal to accept in this attribute, • the greatest code in the attribute 'networkSubsetCodeRangeEnd' and • SHALL not instantiate attribute 'network SubsetCode'. The terminal SHALL interpret all the code values between the smallest and the greatest code as values to be accepted. Applicable only when "type" == 1	integer
networkSubsetCodeRangeEnd	A	NO/ TM	0..1	This attribute signals the end of the range of Network Subset Codes as specified above. Applicable only when "type" == 1	integer
serviceProviderCode	A	NO/ TM	0..1	Service Provider Code as specified by [3GPP TS 22.022]. Applicable only when "type" == 1	byte
corporateCode	A	NO/ TM	0..1	Corporate Code as specified by [3GPP TS 22.022]. Applicable only when "type" == 1	byte
nonSmartCardCode	A	NO/ TM	0..1	Value of BSMFilterCode when "type" == 2	string
Name	E3	NM/TM	1..N	Provides a user readable name for the BSM_Selector, possibly in multiple languages. The language is expressed using built-in XML attribute xml:lang with this element. This element can be used to provide information to the user so he can select the BSMSelector the terminal has to use.	string
RoamingRule	E3	NO/TM	0..N	Entry specifying the RoamingRule associated with BSMSelector.	complexType as defined in section 5.7.1.3 of [BCAST10-Services]
DescriptorE	E1	NM/	1..N	An entry in the Service Guide Delivery	

ntry		TM		Descriptor. Contains the following elements: GroupingCriteria, Transport, AlternativeAccessURL, ServiceGuideDeliveryUnit	
GroupingCriteria	E2	NM/ TM	0..1	Specifies the criteria for grouping Service Guide fragments in this Service Guide DescriptorEntry. If several criteria for grouping are present at the same time, all those grouping criteria apply to the set of Service Guide fragments in this Service Guide DescriptorEntry. Please note the same fragment may be declared in multiple DescriptorEntry of the same SGDD in case this fragment can meet multiple grouping criteria. Contains the following elements: TimeGroupingCriteria GenreGroupingCriteria BSMSelector ServiceCriteria The 'GroupingCriteria' MAY be present as sub-element of 'DescriptorEntry' covering all fragments in this DescriptorEntry. Further, the 'GroupingCriteria' MAY be present as sub-element of 'Fragment' assigning the criteria to a particular fragment. When the 'GroupingCriteria' is present on both of the mentioned levels, the terminal SHALL consider the fragment-level 'GroupingCriteria' providing additional (not overriding) grouping criteria for the fragment.	
TimeGroupingCriteria	E3	NM/ TM	0..1	Specifies the period of time this DescriptorEntry describes. (For example: declares a certain subgroup of valid Service Guide fragments for next 2 hours). This field contains the 32bits integer part of an NTP time stamp. Contains the following attributes: startTime, endTime	
startTime	A	NM/ TM	1	Start of the time period of TimeGroupingCriteria. This field contains the 32bits integer part of an NTP time stamp. Note: this is different than fragment validity time.	unsignedInt
endTime	A	NM/ TM	1	End of the time period of TimeGroupingCriteria. This field contains the 32bits integer part of an NTP time stamp. Note: this is different than fragment validity time.	unsignedInt
GenreGroupingCriteria	E3	NM/ TM	0..1	Specifies the classification of the services/content associated with the fragments in this Service Guide Delivery Unit (e.g. comedy, action, drama).	string

				The OMA BCAST Service Guide allows describing the format of the Genre element in the Service Guide in two ways: • The first way is to use a free string • The second way is to use the "href" attributes of the Genre element to convey the information in the form of a controlled vocabulary (classification scheme as defined in [TVA-Metadata] or classification list as defined in [MIGFG]). The built-in XML attribute xml:lang MAY be used with this element to express the language. The Network MAY instantiate several different sets of 'Genre' element, using it as a free string or with a 'href' attribute. The Network SHALL ensure the different sets have equivalent and non-conflicting meaning, and the terminal SHALL select one of the sets to interpret for the end-user. Contains the following attributes: type href	
type	A	NO/ TO	0..1	This attribute signals the level of this 'Genre' element. The following values are allowed: "main" "secondary" "other"	string
href	A	NO/ TO	0..1	This attribute signals the controlled vocabulary used for this 'Genre' element. If this attribute is supported, terminals MAY support the following classification schemes according to [TVA-Metadata]: • for values of the 'type' attribute equal to "main" or "secondary", the terminal MAY support levels 1-4 of the TV Anytime ContentCS classification scheme identified by the classificationSchemeURI urn:tva:metadata:cs:ContentCS:2005 as defined in Annex A.8 of [TVA-Metadata] • for a value of the 'type' attribute equal to "other", the terminal MAY support levels 1-3 of the TV Anytime IntendedAudienceCS classification scheme identified by the classificationSchemeURI urn:tva:metadata:cs:IntendedAudienceCS:2005 as defined in Annex A.11 of [TVA-Metadata]. When the IntendedAudienceCS is provided	anyURI

				simultaneously with an instantiation of the 'TargetUserProfile', the two SHALL have equivalent meaning. • The network SHALL use the following URI syntax to signal terms from classification schemes: <classificationSchemeURI> ":" <termID> If this attribute is supported, terminals MAY also support the classification from [MIGFG]. In that case: • This classification SHALL be signalled with the URI "http://www.loc.gov/rr/mopic/miggen.html" • The value carried in the 'Genre' element SHALL be used to convey the actual value of the classification as given in [MIGFG] • The Network MAY use values "main" and "secondary" of the 'type' attribute so as to provide an ordering of two classification applying to the same Service. Other Classification Schemes MAY be signalled with the 'href' attribute, however how they are used is out of scope of this specification. For types 'main' and 'secondary', if this attribute is instantiated, the element 'Genre' SHALL be an empty string and the xml:lang attribute SHALL NOT be used. If this attribute is not instantiated, the 'Genre' element SHALL be a free string.	
BSMSelector	E3	NM/ TM	0..N	Specifies the BSM associated with the fragments in this Service Guide Delivery Unit by referencing a BSMSelector structure declared above. Contains the following attribute: idRef	
idRef	A	NM/TM	1	Reference to the identifier of the BSMSelector declared within the 'BSMList' above.	anyURI
ServiceCriteria	E3	NM/TM	0..1	Allows to group fragments by service. The value of this field is the fragment ID of the 'Service' fragment related to that service.	anyURI
Transport	E2	NM/ TM	0..1	The pointer to the transport session delivering the Service Guide fragments within Service Guide Delivery Units announced in this DescriptorEntry. Contains the following attributes: ipAddress, port, srcIpAddress, transmissionSessionID, versionIDLength, hasFDT	

ipAddress	A	NM/ TM	1	Destination IP address of the target delivery session	string
port	A	NM/ TM	1	Destination port of target delivery session	unsignedShort
srcIpAddress	A	NM/ TM	0..1	Source IP address of the delivery session	string
transmissionSessionID	A	NM/ TM	1	This is the Transmission Session Identifier (TSI) of the session at ALC/LCT level	unsignedShort
versionIDLength	A	NO/ TO	0..1	Indicates the number of least significant bits representing the version ID in the transportObjectID, when Split TOI is used. If this element is omitted, the terminal assumes Split-TOI is not used.	unsignedLong
hasFDT	A	NO/ TM	0..1	If FDT is transmitted in the transport session delivering the Service Guide fragments, this attribute SHALL be set to "true". Otherwise this attribute SHALL be set to "false". The default value of this attribute is "true". If this element is set to "false", • the FEC parameters related to transport objects delivering SGDUs in the transport session SHALL be signalled using EXT_FTI [RFC 3926]. • the optional compression of SGDUs SHALL be signalled using EXT_CENC [RFC 3926].	boolean
AlternativeAccessURL	E2	NM/ TM	0..N	Alternative URL for retrieving the Service Guide Delivery Units (SGDUs) via the interaction channel. Note: this sub-element is typically present in the case of interactive delivery of the SG	anyURI
ServiceGuideDeliveryUnit	E2	NM/ TM	1..N	A group of fragments. Contains the following attributes: transportObjectID, contentLocation, validFrom, validTo Contains the following element: Fragment	
transportObjectID	A	NM/ TM	1	The transport object ID of the Service Guide Delivery Unit carrying the declared fragments within this group. If 'hasFDT' is assigned with value 'true', then the value of 'transportObjectID' SHALL match the value of the TOI paired in the FDT instance with the 'Content-Location' having as its value the value of the 'contentLocation' attribute below.	unsignedInt
contentLocation	A	NM/TM	1	This is the location of the Service Guide Delivery Unit. It corresponds to the 'Content-Location' attribute in the FDT.	anyURI
validFrom	A	NM/ TM	0..1	The first moment of time this group of Service Guide fragments is valid. This field contains the	unsignedInt

				32bits integer part of an NTP time stamp. Note: If this attribute is not present, 'validFrom' attribute MUST be present in the 'Fragment' sub-element.	
validTo	A	NM/ TM	0..1	The last moment of time this group of Service Guide fragments is valid. This field contains the 32bits integer part of an NTP time stamp. Note: If this attribute is not present, 'validTo' attribute MUST be present in the 'Fragment' sub-element.	unsignedInt
Fragment	E3	NM/ TM	1..N	Declaration of Service Guide fragment. If the fragment is available over the broadcast channel it MUST be present here. If the fragment is available over the interaction channel it MAY be present here. Contains the following attributes: transportID, id version validFrom validTo fragmentEncoding fragmentType Contains the following element: GroupingCriteria	
transportID	A	NM/ TM	0..1	The identifier of the announced Service Guide fragment to be used in the Service Guide Delivery Unit header. Note: if the SG is delivered over the broadcast channel only, this element MUST be present	unsignedInt
id	A	NM/ TM	1	The identifier of the announced Service Guide fragment.	anyURI
version	A	NM/ TM	1	The version of the announced Service Guide fragment. Note: The scope of the version is limited to the given transport session. The value of version turn over from 2^32-1 to 0.	unsignedInt
validFrom	A	NM/ TM	0..1	The first moment when this fragment is valid. If not given, the validity is assumed to have started at some time in the past. This field contains the 32bits integer part of an NTP time stamp. Note: If this attribute is present and 'validFrom' attribute of 'ServiceGuideDeliveryUnit' is also present, the value of this attribute overrides the value of 'ServiceGuideDeliveryUnit' attribute 'validFrom'.	unsignedInt
validTo	A	NM/ TM	0..1	The last moment when this fragment is valid. If not given, the validity is assumed to end in undefined time in the future. This field contains the 32bits integer part of an NTP time stamp. Note: If this attribute is present and 'validTo' attribute of 'ServiceGuideDeliveryUnit' is also	unsignedInt

				present, the value of this attribute overrides the value of 'ServiceGuideDeliveryUnit' attribute 'validTo'.	
fragmentEncoding	A	NM/TM	1	Signals the encoding of a Service Guide fragment, with the following values: 0 – XML encoded OMA BCAST Service Guide fragment 1 – SDP fragment 2 – MBMS User Service Description as specified in [26.346] (see 5.1.2.4, SessionDescriptionReference) 3 – XML encoded Associated Delivery Procedure as specified in [BCAST10-Distribution] section 5.3.4. 4-127 – reserved for future BCAST extensions 128-255 – available for proprietary extensions	unsignedByte
fragmentType	A	NM/TM	0..1	This field signals the type of an XML encoded BCAST Service Guide fragment, with the following values: 0 – unspecified 1 – 'Service' Fragment 2 – 'Content' fragment 3 – 'Schedule' Fragment 4 – 'Access' Fragment 5 – 'PurchaseItem' Fragment 6 – 'PurchaseData' Fragment 7– 'PurchaseChannel' Fragment 8 – 'PreviewData' Fragment 9 – 'InteractivityData' Fragment 10-127 – reserved for BCAST extensions 128-255 – available for proprietary extensions This attribute SHALL be present in case 'fragmentEncoding'=0. Default: 0	unsignedByte
GroupingCriteria	E4	NM/ TM	0..1	Specifies the criteria for grouping this Service Guide fragment. If several criteria for grouping are present at the same time, all those grouping criteria apply to this fragment. Contains the following elements: TimeGroupingCriteria BSMSelector The 'GroupingCriteria' MAY be present as sub-element of 'DescriptorEntry' covering all fragments in this DescriptorEntry. Further, the 'GroupingCriteria' MAY be present as sub-element of 'Fragment' assigning the criteria to a particular fragment. When the 'GroupingCriteria' is present on both of the mentioned levels, the terminal SHALL consider the fragment-level 'GroupingCriteria' providing additional (not	

				overriding) grouping criteria for the fragment.	
TimeGroupingCriteria	E5	NM/ TM	0..1	Specifies the period of time as a grouping criteria for the fragment. (For example: declares a certain subgroup of valid Service Guide fragments for next 2 hours). This field contains the 32bits integer part of an NTP time stamp. Contains the following attributes: startTime endTime If a 'TimeGroupingCriteria' element exists at DescriptorEntry level which is applicable to this fragment, the time interval defined here at 'Fragment' level MUST NOT exceed the time interval defined at 'DescriptorEntry' level.	
startTime	A	NM/ TM	1	Start of the time period of TimeGroupingCriteria. This field contains the 32bits integer part of an NTP time stamp. Note: this is different than fragment validity time.	unsignedInt
endTime	A	NM/ TM	1	End of the time period of TimeGroupingCriteria. This field contains the 32bits integer part of an NTP time stamp. Note: this is different than fragment validity time.	unsignedInt
BSMSelector	E5	NM/ TM	0..N	Specifies the BSM associated with this fragment by referencing a BSMSelector structure declared above. Contains the following attribute: idRef	
idRef	A	NM/TM	1	Reference to the BSMSelector declared within the BSMList structure above.	anyURI
PrivateExt	E1	NO/ TO	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	NO/TO	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.4.1.5.3 Grouping of Service Guide

One can divide the set of fragments comprising a Service Guide into subsets. These subsets, also referred to as groups, are formed by first selecting a criterion (or possibly criteria). Examples of a criterion are intervals of time, BSM codes and identifiers of 'Service' fragments. Once a criterion is selected, each subset (or group) to be formed is assigned a value from the domain of the criterion. The contents of each subset are then determined by comparing the value of the criterion represented by the subset to each of the fragments of the Service Guide. If and only if the values of the subset and the fragment match, the fragment is placed into the subset

For example, one could have a subset representing all the fragments representing the contents of the SG today from 10 a.m. to 12 a.m. In such a case the grouping criteria is an interval of time, the value represented by the subset is the interval from 10 o'clock to 12 o'clock and any fragment representing the contents of the Service Guide within that time frame belongs to the subset.

The network MAY

- use multiple grouping criteria simultaneously,

- form overlapping (i.e. mutually non-exclusive) subsets (or groups) and
- place the declaration(s) of the subsets or even a single subset into multiple SGDDs.

But the network SHALL NOT

- place a fragment into a subset if the fragment contains a reference to a fragment not present in the same subset.

This restriction introduces the constraint of consistency on the Service Guide subsets (or groups). A Service Guide subset is consistent when no fragment from that subset references any fragment that is not in the said subset. If and only if all the subsets formed by the network comply with this rule, is the Service Guide considered consistent.

5.4.1.5.4 Availability of the SG fragments over broadcast and over interactive channel

In the 'DescriptorEntry' element of the 'SGDD' one can declare fragments to be available explicitly via broadcast and/or via interactive channel. The availability over broadcast channel is declared using the 'Transport' element of the 'DescriptorEntry' whereas the availability over interactive channel is declared using the 'AlternativeAccessURL'. The presence of these elements in the 'DescriptorEntry' SHALL be interpreted by the terminal as follows:

- If only 'Transport' element is instantiated the fragments declared in the corresponding 'DescriptorEntry' are available only over the broadcast channel according to the information in the 'Transport' element.
- If only 'AlternativeAccessURL' elements are instantiated the fragments declared in the corresponding 'DescriptorEntry' are available only over the interactive channel from the locations designated by the values of the 'AlternativeAccessURL' elements.
- If both the 'Transport' element and at least one 'AlternativeAccessURL' element are instantiated the fragments declared in the corresponding 'DescriptorEntry' element are available over both the broadcast and the interactive channels.

5.4.2 Delivery over the Broadcast Channel

Over the Broadcast Channel, interface SG-5, the Service Guide is delivered using broadcast file delivery sessions. The network places the fragments of the Service Guide into one or more SGDUs and constructs one or more SGDDs to represent the contents of the Service Guide as well as the division of the fragments into the SGDUs. The SGDD(s) and the SGDU(s) are placed into file delivery session(s) to be transported as transport objects, TOs. While the SGDUs can be transported using one or more file delivery sessions, the SGDDs are provided using only of the sessions, namely the Service Guide Announcement Channel as defined in section 5.4.1.5.1.

The network SHALL use FLUTE [RFC 3926] as the protocol for broadcast delivery of the Service Guide and the network SHALL provide FDT Instances in the Service Guide Announcement Channel and the network MAY provide FDT Instances in the other sessions carrying the Service Guide.

The following enhancements apply for the case when the file information is conveyed in the Service Guide or in a file delivery table:

- SG-D in BSD/A MAY apply the "Compact No-Code FEC scheme" [RFC 3695] (FEC Encoding ID 0, also known as "Null-FEC").
- SG-D in BSD/A MAY utilize the split-TOI scheme as specified in section 5.4.2.1.3 in conjunction with FLUTE, for signalling the identifier and version of any transported object (e.g. the Service Guide Delivery Unit or Service Guide Delivery Descriptor).
- SG-D in BSD/A MAY utilize the scheme as specified in section 5.4.2.1.3 in conjunction with FLUTE, for signalling the identifier and version of the Service Guide Delivery Unit.

In order for the terminals to distuinguish the SGDDs and SGDUs from other transport objects the network SHALL set the ‘Content-Type’ attribute of the ‘File’ element in the FDT Instances

- to “application/vnd.oma.bcast.sgdd+xml” for SGDDs and

- to “application/vnd.oma.bcast.sgdu” for SGDUs.

As there is no signalling whether the network uses FDT Instances in the Service Guide delivery sessions other than the Service Guide Announcement Channel, the terminal

- SHALL assume that the Transport Object Identifier, TOI, zero is reserved for the FDT Instances.
- And the network SHALL not use the TOI zero for any types of files than FDT Instance.

The network SHALL signal the Forward Error Correction, FEC, parameters for the transport objects in the Service Guide delivery sessions using one of the mechanisms defined in FLUTE [RFC 3926] and the terminal SHALL support all these mechanisms.

5.4.2.1 Signaling Changes in the Service Guide over Broadcast Channel

In the following, the way of signaling changes in Service Guide fragments is specified. The changes in the Service Guide are signaled through the change in the transmitted SGDUs which consequently cause a change in the transmitted SGDDs. Observing these changes, the terminal SHALL be able to determine the change. However, this specification does not specify the normative terminal behavior for this. Informative examples for four cases of localizing changes and achieving their discovery are outlined in section 5.5.

5.4.2.1.1 Signalling Addition of a New SGDU

Upon addition of a newn SGDU to be delivered on the Service Guide Delivery Channel, a new TOI is allocated for the delivery of the SGDU.

- If FLUTE is used on the Service Guide Delivery Channel, the allocated TOI SHALL be introduced in the FDT associated with Service Guide Delivery Channel.
- If ALC is used on the Service Guide Delivery Channel, it is assumed that the Terminal detects the change of TOI by observing the session.

Further, the allocated TOI SHALL be introduced in the SGDD on the Service Guide Announcement Channel. Consequently, the TOI of SGDD itself SHALL change. This SHALL be indicated through introducing the new TOI of the SGDD in the FDT associated with Service Guide Announcement Channel. While using FLUTE, the filenames associated with each Service Guide Delivery Descriptor withinto the FDT SHALL be set so that the terminal can use the FDT information to identify the transported Service Guide Delivery Descriptor, and its version.

- The ‘Content-Location’ attribute of the ‘File’ element within the FDT is used for this purpose. The Service Guide Delivery Descriptor is identified by its SGDDid (see section 5.4.1.5.2), which SHALL be used as a unique URI for the ‘Content-Location’ attribute.
- The version change of the Service Guide Delivery Descriptor is signalled based on the TOI and FDT Instance ID as specified in section 5.2.4 in [BCAST10-Distribution].

In addition, if Split-TOI scheme is used, the network MAY signal version relation between a removed SGDD and a newly inserted SGDD by allocating the TOI for the newly inserted SGDD, i.e. by taking the Most Significant Bits of the removed SGDD’s TOI and changing only the Least Significant Bits of the TOI.

5.4.2.1.2 Signalling Change in SGDUs

Upon change in SGDUs one or more SGDUs are inserted to or removed from the Service Guide Delivery Channel. The TOIs corresponding to the removed SGDUs SHALL be disassociated with the SGDDs which the SGDUs were associated with before, and the TOIs corresponding to the inserted SGDUs SHALL be signaled according to chapter 5.4.1.2.1.

- In addition, if Split-TOI scheme is used, the network MAY signal version relation between a removed SGDU and a newly inserted SGDU by allocating the TOI for the newly inserted SGDU, i.e. taking the Most Significant Bits of the removed SGDU’s TOI and changing only the Least Significant Bits of the TOI.

- In addition, if Split-TOI scheme is used, the network MAY signal version relation between a removed SGDD and a newly inserted SGDD by allocating the TOI for the newly inserted SGDD by, i.e. taking the Most Significant Bits of the removed SGDD's TOI and changing only the Least Significant Bits of the TOI.

5.4.2.1.3 Split-TOI scheme

To provide a mechanism for terminals to easily track Service Guide updates when the Service Guide is delivered over broadcast channel using ALC or FLUTE, this section specifies a method to use the LCT TOI field of a transported SGDD/SGDU to indicate its identifier and its version, so that terminals can track the delivery of new versions of this SGDD/SGDU without parsing its payload.

When FLUTE is used, the Split TOI mechanism especially reduces the need to distribute FDT Instances functionally restricted to signal new versions of SGDDs/SGDUs (i.e. FDT Instances not describing new SGDDs/SGDUs, or not updating the "Expires" time for some SGDDs/SGDUs, etc.). Apart from the potential reduction of distributed FDT Instances, this mechanism still relies on the distribution of FDT Instances, and not does relieve the FD-C of its parsing obligations with regard to any received FDT Instances.

The LCT TOI field is 32*O + 16*H bits in length where the Transport Object Identifier flag (O) length is 2 bits and the Half-word flag (H) length is 1 bit. The maximal length of the TOI is therefore 112 bits (i.e. 14 bytes).When a version identifier is assigned to a transported object through the LCT header, the TOI field is split into two parts: the first part (Most Significant Bits) called Object ID part is allocated to the identification of SGDU/SGDD, the second part (Less Significant Bits) called Version ID part is allocated to the version of the identified SGDU/SGDD. The terminal can track SGDU/SGDD updates based on the changes in the Version ID part of the TOI.

The receiver detects whether the TOI is split or not through in-band signalling as specified below:

- for an SGDD/SGDU transported in a FLUTE session, the indication of a split TOI SHALL be signalled in all FDT Instances declaring this SGDD/SGDU by the inclusion of "Version-ID-Length" attribute either in <FDT-Instance> top-level element or in the <File> element describing this SGDD/SGDU. The former case signals that the TOI of each SGDD/SGDU described in the FDT Instance is split, using "Version-ID-Length" specified at <FDT-Instance> element level unless overriden by "Version-ID-Length" provided at <File> element level.
- for an SGDU transported in an ALC session, the indication of split TOI SHALL be signalled in all SGDD structures describing this SGDU, by the inclusion of "versionIDLength" attribute in the element declaring the TOI for this SGDU (i.e. <ServiceGuideDeliveryUnit> element).

Whatever use is made of Split TOI scheme in an SG delivery session (such as TOIs split possibly with different Version ID lengths, mixed with TOIs not split), the uniqueness of each LCT TOI value generated in the session as well as the one-to-one mapping between TOIs and transport objects SHALL still apply.

Besides, for all SGDDs with a split TOI delivered in the same FLUTE session:

- there SHALL be a one-to-one mapping between SGDD 'id' (URI) and Object ID part in LCT TOI.
- SGDD "version" number SHALL be equal to Version ID part in LCT TOI.

The network MAY utilize the Split-TOI scheme for signalling the identifier and version of transported object, and terminals MAY be able to interpret the split TOI field in ALC header.

5.4.3 Delivery over the Interaction Channel

If a terminal has access to the Interaction Channel, then it SHALL support the mechanisms for accessing the Service Guide over Interaction Channel as defined by the provisions of this section.

The following gives a brief overview on the ways of requesting Service Guide over the Interaction Channel in the light of use cases enabled:

1. Terminal can request Service Guide fragments by their identifiers. This requires that the terminal knows the identifiers of the fragments prior to request. Consequently, the terminal can use this request to update the version of

the fragments it already has or to retrieve fragments declared in the Service Guide Delivery Descriptor it had acquired earlier.

2. Terminal can request Service Guide Delivery Descriptors by their identifiers. This requires that the terminal knows the identifiers of the Service Guide Delivery Descriptors prior to request. Consequently, the terminal can use this request to update the version of the Service Guide Delivery Descriptors.

3. Terminal can specify within the request whether it wants responses as SGDDs or SGDUs. This way the terminal can either get the declarations of fragments, the fragments itself or both.

4. Terminal can request Service Guide Delivery Descriptors by using the grouping criteria present in Service Guide Delivery Descriptors as request criteria. This way the terminal can request all Service Guide Delivery Descriptors fulfilling the given criteria. If the terminal also requested to have the fragments, it will receive the fragments that are available over Interaction Channel.

5. Terminal can request Service Guide fragments by using the attributes / elements present in Service Guide fragments as request criteria. This way the terminal can request all Service Guide fragments fulfilling the given criteria. If the terminal also requested to have the SGDDs, it will receive the SGDDs that declare the fragments.

6. Terminal can request Service Guide fragments and/or Service Guide Delivery Descriptors without any specific criteria or identifiers. This way terminal can request “any” view of Service Guide and let network to decide which set to provide to terminal.

7. Terminal can request all Service Guide fragments and/or Service Guide Delivery Descriptors. This way terminal can request the widest possible view to Service Guide.

In the use cases 3-7 the terminal should provide at least one BSMSelector value as a part of the query to limit the response only to SG information that is associated with the terminal’s affiliated BSM(s).

5.4.3.1 Rules applicable to all requests and responses

When requesting either Service Guide fragments or Service Guide Delivery Descriptors over Interaction Channel, the terminal and network SHALL comply with the following rules:

- The terminal SHALL originate requests. The network SHALL respond to requests.
- The request SHALL be made using ‘POST’ method of HTTP/1.1
- The parameters associated with the request SHALL be communicated as key-value pairs following the conventions defined in section 17.13 of [HTML4.01] for submitting HTML form data by the ‘POST’ method using the "application/x-www-form-urlencoded" encoding type. More specifically, once encoded as "application/x-www-form-urlencoded", the parameters to be passed from terminal to system SHALL be communicated in the ‘message-body’ of HTTP/1.1 ‘Request’ message as defined in section 5 of [RFC 2616].
    - Within a single request, the terminal MAY include multiple key-value pairs. As defined by [HTML4.01] these key-value pairs SHALL be delimited by an ‘&’.
    - Within the request, the terminal MAY specify the requested format of response.
        - When the terminal requests only Service Guide Delivery Descriptors the ‘message-body’ of HTTP/1.1 request SHALL be prefixed with “type=sgdd”.
        - When the terminal requests only Service Guide Delivery Units the ‘message-body’ of HTTP/1.1 request SHALL be prefixed with “type=sgdu”
        - When the terminal requests Service Guide Delivery Descriptors and all Service Guide fragments declared within the Service Guide Delivery Descriptors that are available over Interaction Channel, the ‘message-body’ of HTTP/1.1 request SHALL be prefixed with “type=sgdd+sgdu”.
- The response to the request SHALL be HTTP/1.1 response with status ‘200 OK’.
- The response SHALL have HTTP header field ‘Content-Type’ set to “text/xml”

- The HTTP payload of response SHALL contain one instance of element "SGResponse". See 5.4.3.1.1 for the definition of element "SGResponse".
    - The element "SGResponse" carries the status code associated with the response.
    - If Service Guide Delivery Descriptors are returned with the response, those SHALL be carried within the element "SGResponse".
- The element "SGResponse" MAY be immediately followed by one Service Guide Delivery Unit. If Service Guide fragments are returned with the response, those SHALL be encapsulated in that Service Guide Delivery Unit.

5.4.3.1.1 Definition of element "SGResponse"

Name	Type	Category	Cardinality	Description	Data Type
SGResponse	E			Delivers the status of response to interactive Service Guide request. Also contains Service Guide Delivery Descriptors associated with the response. Contains the following attribute: status Contains the following elements: ServiceGuideDeliveryDescriptor PrivateExt	
status	A	M	1	Declares status of the interactive Service Guide delivery response using 'GlobalStatusCode' defined in the section 5.1.4 of [BCAST10-Services]. The following status codes SHALL NOT be used: 001-006, 009-016, 019, 022, 024-027	unsigned Byte
ServiceGuideDeliveryDescriptor	E1	O	0..N	Service Guide Delivery Descriptor. See section 5.4.1.5.2.	complexType
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.4.3.2 Unspecific request for retrieving service guide over Interaction Channel

If the terminal supports unspecific request for retrieving service guide over Interaction Channel, the terminal SHALL request Service Guide fragments and Service Guide Delivery Descriptors over the Interaction Channel as follows:

- When terminal requests Service Provider's default view to Service Guide, the 'message-body' of HTTP/1.1 request message SHALL contain no data.
- The 'Request-URI' of HTTP POST request SHALL be set to the Service Guide entry point address (URL).
- The response of the request MAY contain Service Guide Delivery Descriptors, Service Guide fragments or both.

When the response contains Service Guide fragments, the returned set of fragments SHALL be consistent as specified in 5.4.1.5.3.

5.4.3.3 Requests for specific Service Guide Delivery Descriptors

If the terminal supports requests for specific Service Guide Delivery Descriptors, the terminal SHALL request specific Service Guide Delivery Descriptors over the Interaction Channel as follows:

- When terminal requests individual Service Guide Delivery Descriptors by their identifiers, the 'message-body' of HTTP/1.1 request message SHALL contain one or several key-value pairs, using "sgddID" as the key and the attribute 'id' of the requested SGDD as the value .
- When terminal requests Service Guide Delivery Descriptors with criteria other than SGDD identifier, the 'message-body' of HTTP/1.1 request message SHALL contain one or severa key-value pairs, using <key> as the key representing the criteria and the <value> as the value from the domain of the criteria. If several key-value pairs are given, they are combined as follows
    - key-value pairs having the same key are combined using OR logic, i.e. the expected reply are Service Guide Delivery Descriptors that satisfy at least one of the given criteria. The group of OR-combined keys is in the next step below treated as one entity.
    - Key-value pairs (or groups of pairs having the same key) having different keys are combined using AND logic, i.e. the expected reply are Service Guide Delivery Descriptors that satisfy all given criteria.
- Furthermore, terminal SHALL support the following assignments for the <key> and <value> fields:
    - <key> : "tgc-start"
      <value> : Attribute 'startTime' associated with element 'TimeGroupingCriteria' used within SGDD
      Meaning : Terminal requests SGDD grouped with 'TimeGroupingCriteria', whose 'startTime' equals to <value>.
      Note : This <key> SHALL be used if and only if <key> "tgc-end" is used.
    - <key> : "tgc-end"
      <value> : Attribute 'endTime' associated with element 'TimeGroupingCriteria' used within SGDD
      Meaning : Terminal requests SGDD grouped with 'TimeGroupingCriteria', whose 'endTime' equals to <value>.
      Note : This <key> SHALL be used if and only if <key> "tgc-end" is used.
    - <key> : "ggc"
      <value> : Value of element 'GenreGroupingCriteria' used within SGDD.
      Meaning : Terminal requests SGDD grouped with 'GenreGroupingCriteria', whose value equals to <value>.
    - <key> : "srvc"
      <value> : Value of element 'ServiceCriteria' used within SGDD.
      Meaning : Terminal requests SGDD grouped with 'ServiceCriteria', whose value equals to <value>.
    - <key> : "bsms"
      <value> : Value of element 'BSMSelector' used within SGDD. The value is given as the following structure:

      If 'type' equals "1":
      "1+"<mobileCountryCode>"+"< mobileNetworkCode>"+"<networkSubsetCode>"+"
      <networkSubsetCodeRangeStart>"+"< networkSubsetCodeRangeEnd >"+"
      <serviceProviderCode>"+"<corporateCode>"

      If 'type' equals "2":
      "2+"<nonSmartCardCode>

      Meaning : Terminal requests SGDD grouped with 'BSMSelector', whose value equals to <value>.

      Note : This <key> SHOULD always be used when requesting SGDD(s) with a key other than "sgddID".

- <key> : "complete"
  <value> : "true"
  Meaning : Terminal requests a complete set of SGDDs.

  Note : This key SHOULD NOT be combined with criteria i.e. keys, restricting the set of SGDDs.

- <key> : "user"
  <value> : User id.
  Meaning : Terminal requests a set of SGDDs personalized to the user associated with user identification of <value>.

- The 'Request-URI' of HTTP POST request SHALL be set to the Service Guide entry point address (URL).
- The response of the request MAY contain Service Guide Delivery Descriptors, Service Guide fragments or both.

5.4.3.4 Requests for specific Service Guide fragments

If the terminal supports requests for specific Service Guide fragments, the terminal SHALL request specific Service Guide fragments over the Interaction Channel as follows:

- When terminal requests individual Service Guide fragments by their identifiers, the 'message-body' of HTTP/1.1 request message SHALL contain key-value pair, using "fragmentID" as the key and the attribute 'id' of the requested fragment as the value.
- When terminal requests Service Guide fragments associated to an SGDD, the 'message-body' of HTTP/1.1 request message SHALL contain key-value pair, using "sgddID" as the key and the attribute 'id' of the SGDD as the value.
- When terminal requests Service Guide fragments with criteria other than the fragment identifier or SGDD identifier, the 'message-body' of HTTP/1.1 request message SHALL contain key-value pairs, using <key> as the key representing the criteria and the <value> as the value from the domain of the criteria. These key-value pairs SHALL be delimited by a '&'. Furthermore, terminal SHALL support the following assignments for the <key> and <value> fields:
  - <key> : "globalServiceID"
    <value> : Attribute 'globalServiceID' used within 'Service' fragment
    Meaning : Terminal requests Service Guide fragment associated with 'Service' fragments having 'globalServiceID' equal to <value>.

  - <key> : "globalContentID"
    <value> : Attribute 'globalContentID' used within 'Content' fragment
    Meaning : Terminal requests Service Guide fragment associated with 'Content' fragments having 'globalContentID' equal to <value>.

  - <key> : "validFrom"
    <value> : Attribute 'validFrom' used within Service Guide fragments
    Meaning : Terminal requests Service Guide fragments whose attribute 'validFrom' is greater or equal to the <value>

  - <key> : "validTo"
    <value> : Attribute 'validTo' used within Service Guide fragments
    Meaning : Terminal requests Service Guide fragments whose attribute 'validTo' is less or equal to the <value>

  - <key> : "contentType"
    <value> : Attribute 'contentType' used within 'Content' fragment
    Meaning : Terminal requests Service Guide fragments associated with 'Content' fragments having 'contentType' equal to <value>.

  - <key> : "serviceType"
    <value> : Attribute 'serviceType' used within 'Service' fragment

Meaning : Terminal requests Service Guide fragments associated with 'Service' fragments having 'serviceType' equal to <value>.

- <key> : "genre"
  <value> : Element 'genre' used within Service Guide fragments
  Meaning : Terminal requests Service Guide fragments and fragments associated to them whose element 'genre' equal to <value>.

- <key> : "fragmentEncoding"
  <value> : Possible values listed in Table 3 under 'fragmentEncoding'.
  Meaning : Terminal requests Service Guide fragments that are encoded with encoding scheme defined by <value>.

- <key> : "fragmentType"
  <value> : Possible values listed in Table 3 under 'fragmentType'.
  Meaning : Terminal requests Service Guide fragments of that are of type <value>.

- <key> : "all"
  <value> : "true"
  Meaning : Terminal requests all available Service Guide fragments.

- <key> : "consistent"
  <value> : "true"
  Meaning : Terminal requests consistent set of Service Guide fragments.

- <key> : "bsms"
  <value> : Value of element 'BSMSelector' used within SGDD. The value is given as the following structure:

  If 'type' equals "1":
  "1+"<mobileCountryCode>"+"< mobileNetworkCode>"+"<networkSubsetCode>"+"
  <networkSubsetCodeRangeStart>"+"< networkSubsetCodeRangeEnd >"+"
  <serviceProviderCode>"+"<corporateCode>"

  If 'type' equals "2":
  "2+"<nonSmartCardCode>

  Meaning : Terminal requests Service Guide fragments filtered with 'BSMSelector', whose value equals to <value>.

  Note : This <key> SHOULD always be used when requesting Service Guide fragments with a key other than "fragmentID" or "sgddID".

- <key> : "modified-since"
  <value> : NTP time
  Meaning : Terminal requests a set of fragments modified since the specified point in time.

- <key> : "BroadcastAccess"
  <value> : Attribute 'Type' under element 'BDSType' in Access fregment
  Meaning : Terminal requests Service Guide fragments receivable over the specified Broadcast access

- <key> : "UnicastAccess"
  <value> : Attribute 'type' under element 'UnicastServiceDelivery' in Access fregment
  Meaning : Terminal requests Service Guide fragments receivable over the specified Unicast access

- The following applies for the selection of the target for the request:
  - Upon the terminal requesting an individual Service Guide fragment by its identifier the terminal does this as follows: In case the terminal had acquired an SGDD that declared the fragment in question and the element 'AlternativeAccessURL', the 'Request-URI' of HTTP POST request SHALL be set to 'AlternativeAccessURL'.

  - In any other case, the 'Request-URI' of HTTP POST request SHALL be set to the Service Guide entry point address (URL)

- In the response the requested Service Guide fragments SHALL be encapsulated in a SGDU as defined in section 5.4.1.3.
- If the request contained <key> "consistent" with <value> "true", the returned set of fragments SHALL be consistent as specified in 5.4.1.5.3.

5.5 Service Guide Update and Management

Over time, as services, content and schedules come and go, the metadata associated with Service Guide changes. The terminal needs to be able identify the changes and to manage the updates of Service Guide and associated metadata accordingly. There are two ways the terminal SHALL support: 1) update and management of Service Guide over broadcast; and; 2) update and management of Service Guide over interaction. These ways differ in the specification of delivery method. However, in both of the cases the metadata management is done finally on the level of Service Guide fragments. Hence, the terminal SHALL be able to determine the identification, version and validity of each Service Guide fragment using the respective fields in the Service Guide fragment itself. Further, the terminal SHALL be able to interpret the following cases:

- Fragment to be processed has fragment id that has not been previously received. This means that the terminal has received a new piece of metadata.
- Fragment to be processed has same fragment id and same version than has been previously received. This means that no change is implied. The metadata that was already received is still valid.
- Fragment to be processed has same fragment id but higher version than has been previously received. This means that the metadata related to the fragment in question needs to be updated. If 'validFrom' attribute of the new fragment is not present or indicates some time in past, the terminal SHALL replace the previous metadata with the newly received metadata immediately. If 'validFrom' attribute of the new fragment is present and indicates some time in future, the terminal SHALL store the new fragment and keep using the existing one until the new one gets valid.
- Fragment to be processed has same fragment id but lower version than has been previously received. This means that the previously received metadata related to the fragment in question is more recent than the meatadata in the fragment that was just received. Terminal SHALL discard the new fragment.
- The terminal SHALL handle the possible wrap-around of the version.

Further, overall validity of a Service Guide fragment MAY be expressed with OPTIONAL attributes 'validFrom' and 'validTo'. The terminal SHALL support the interpretation of these attributes in determining whether a given Service Guide fragment is valid on a given moment of time: If the current time is within the time span defined by 'validFrom' and 'validTo' the terminal SHALL consider that the metadata is a valid member of the set of fragments comprising the SG. Outside of the specified validity, the fragment SHALL NOT be considered part of the currently applicable SG (e.g. with respect to consistency), and the fragment information SHOULD NOT be used by the terminal, or rendered to the user at this point of time.

If the set of fragments belonging to the Service Guide are announced using the mechanism defined in section 5.4.1.1, then the terminal MAY assume from the absence of any fragment in the updated version of the SGDDs that the validity of the fragment has ended.

5.5.1 Service Guide Update and Management using the Broadcast Channel

Over the broadcast channel the transmitted Service Guide can be large and consist of several delivery sessions carrying several parts of the Service Guide. Thus, as the total size of Service Guide can be large – both in terms of data amount per fragment as well as in terms of fragments – it is useful to localize the changes to parts of Service Guide. This allows a terminal to identify as narrow changes as possible. Further it allows the terminal to receive the updates and manage data only for the relevant part(s) of Service Guide.

On the other hand, assuming that the terminal has received the Service Guide or a part of Service Guide, the structure of the Service Guide makes it possible for the terminal to determine quickly whether there is relevant information available in the Service Guide, which the terminal does not possess yet. Since the content of the Service Guide can and do change over time,

and one aims to minimize the time needed by the terminals to update their view of the Service Guide, one wants to have a mechanism for declaring the contents of the Service Guide without forcing the terminal to traverse through the entire set of fragments of the Service Guide.

The methods for localized changes and their discovery, and, for determining the well-defined subsets of the Service Guide are specified in the following sections.

5.5.1.1 Localized changes and their discovery (Informative)

The flow of changes and how they propagate in the Service Guide is illustrated in the Figure 3-6 below. This way even the smallest changes in the Service Guide (i.e. change in individual fragments) can be accurately pointed out.

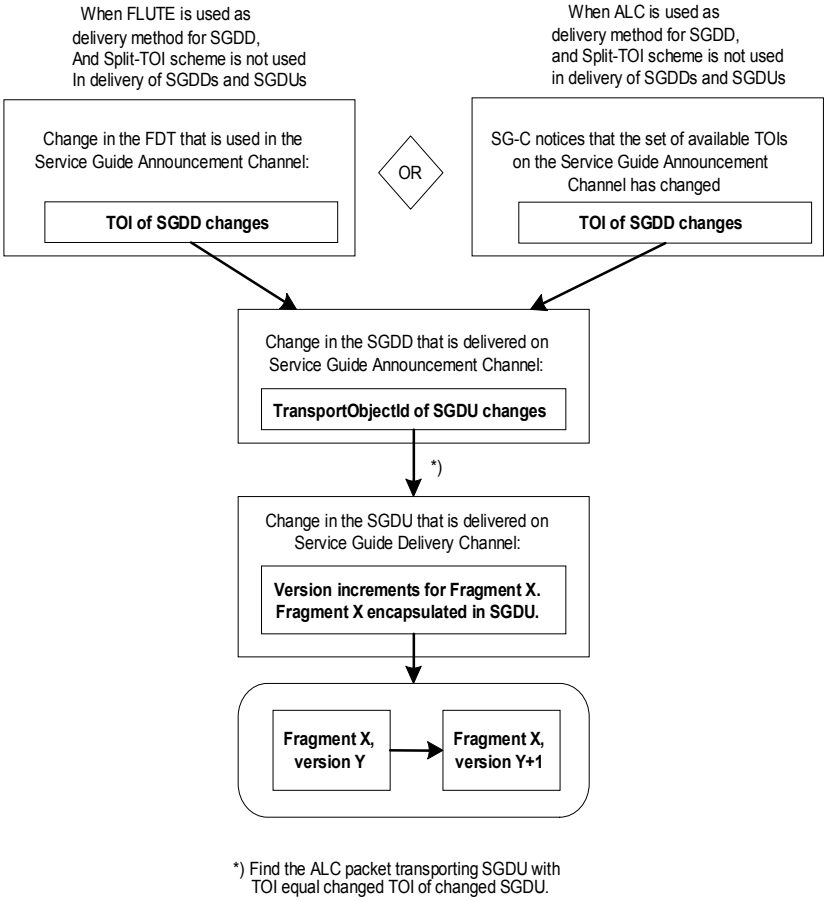


Figure 3: SG fragment change and its propagation on the different levels of Service Guide, when Split-TOI is not used neither for SGDU nor for SGDD

(Note: when FLUTE is used as the delivery of SGDUs, the FDT of the session carrying the SGDUs is also updated when TOI of the SGDU is changed.)

This flowchart in Figure 3 assumes that Split-TOI scheme is not used in the delivery of SGDDs/SGDUs. Here, the change of Fragment X version Y to Fragment X version Y+1 causes a change in the respective SGDU, whose TOI changes on the Service Guide Delivery Channel. Consequently, it causes the declaration of the SGDU in the respective SGDD to change. Further, as the SGDD changes, so does its Transport Object Identifier on the Service Guide Announcement Channel. Finally the SG-C notices this change through observing the change in the FDT table available on the Service Guide Announcement Channel. Correspondingly, if the original corresponding fragment was of interest to the terminal, it will store and parse the associated SGDU. However, if that original fragment was not of interest to the SG-C, then the fragment version update would similarly be of no interest, and the SG-C will disregard the associated SGDU.

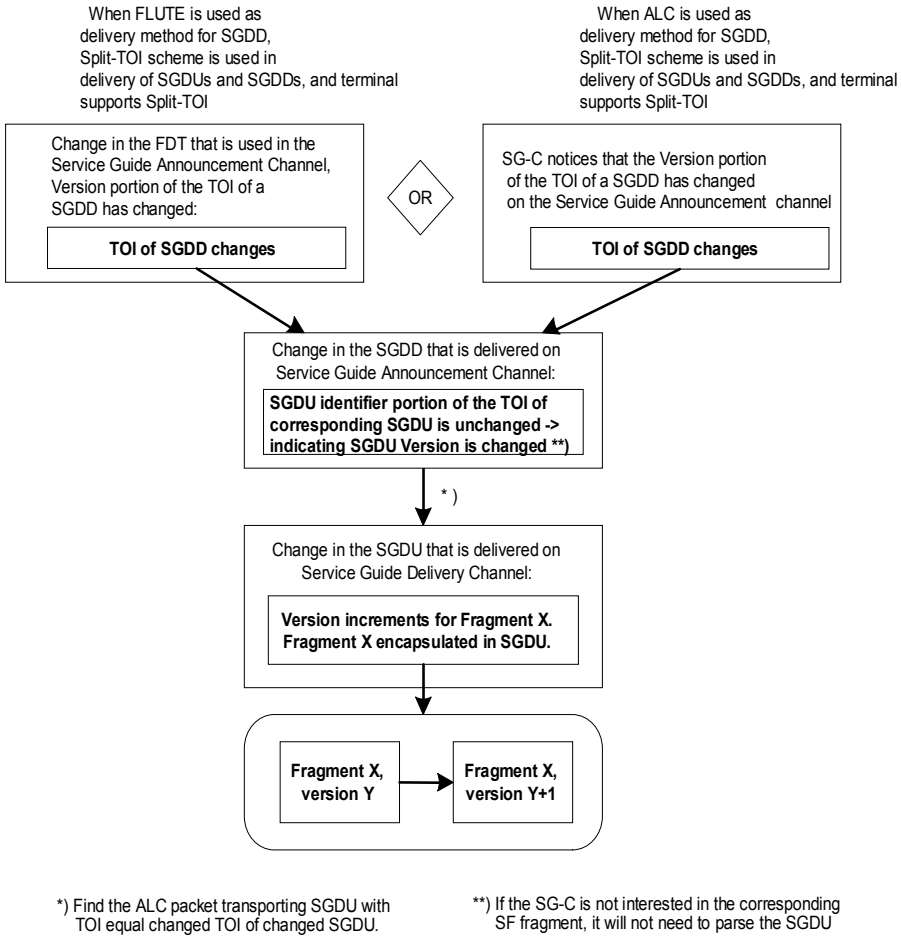


Figure 4: Fragment version change and its detection at ALC packet header level of SGDU/SGDD delivery, when Split-TOI is used

The flowchart in Figure 4 assumes that Split-TOI scheme is used in the delivery of SGDUs/SGDDs. Here, the change of Fragment X version Y to Fragment X version Y+1 causes a change in the respective SGDU, whose Version portion of the TOI is changed on the Service Guide Delivery Channel. Consequently, it causes the declaration of the SGDU in the respective SGDD to change. Further, as the SGDD changes, so does its Transport Object Identifier on the Service Guide Announcement Channel, and particularly the Version portion of the TOI. The SG-C notices this change through observing the change in the FDT table available on the Service Guide Announcement Channel. In addition, the SG-C is able to detect the indicated SG fragment change as a version update. Correspondingly, if the original corresponding fragment was of interest to the terminal, it will store and parse the associated SGDU. However, if that original fragment was not of interest to the SG-C, then the fragment version update would similarly be of no interest, and the SG-C will disregard the associated SGDU.

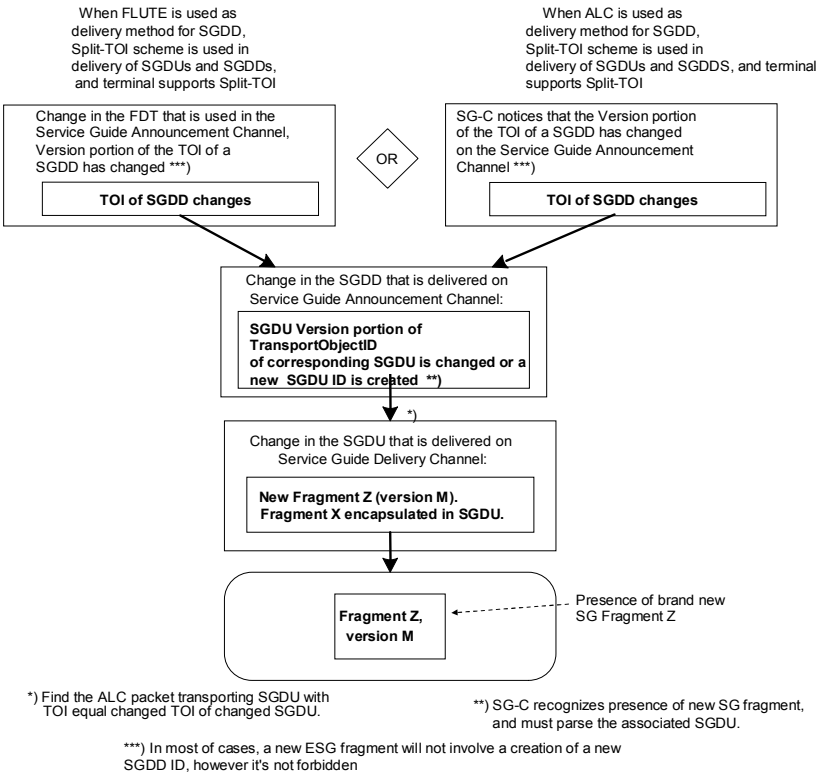


Figure 5: New fragment introduced and its detection at ALC packet header level of SGDU delivery, when Split-TOI is used

The flowchart in Figure 5 assumes that Split-TOI scheme is used in the delivery of SGDUs. Here, the presence of a brand new SG fragment Z causes a change in the respective SGDU, whose Version portion of the TOI is changed on the Service Guide Delivery Channel, or causes a new SGDU identifier portion to be created. Consequently, it causes the declaration of the SGDU in the respective SGDD to change. Further, as the SGDD changes, so does its Transport Object Identifier on the Service Guide Announcement Channel, and particularly the Version portion of the TOI. The SG-C notices this change through observing the change in the FDT table available on the Service Guide Announcement Channel. Correspondingly, to find out about this new fragment, the SG-C will store and parse the associated SGDU.

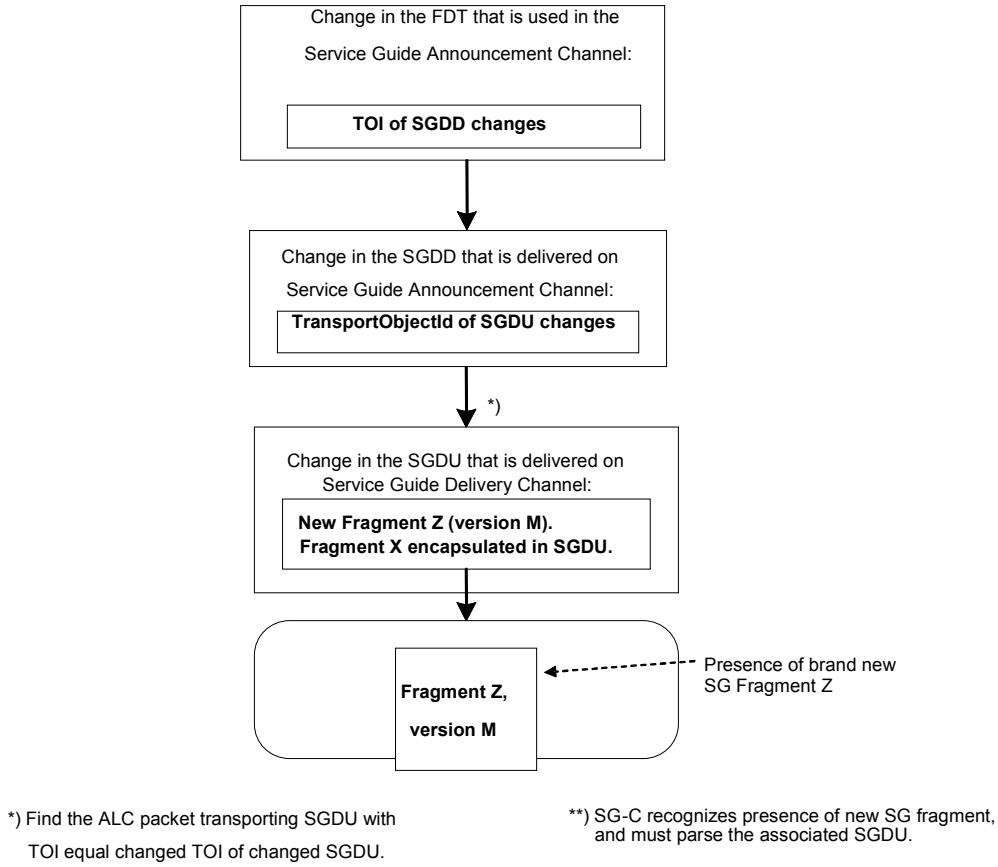


Figure 6: New fragment introduced and its propagation on the different levels of Service Guide, when Split-TOI is not used

This flowchart in Figure 6 assumes that Split-TOI scheme is not used in the delivery of SGDDs/SGDUs. Here, the presence of a brand new SG fragment Z causes a change in the respective SGDU, whose Object identifier portion of the TOI is changed on the Service Guide Delivery Channel. Consequently, it causes the declaration of the SGDU in the respective SGDD to change. Further, as the SGDD changes, so does its Transport Object Identifier on the Service Guide Announcement Channel. Finally the SG-C notices this change through observing the change in the FDT table available on the Service Guide Announcement Channel. As the SG-C in the terminal notices the change in either of these ways, following the chain of propagation allows the SG-C to accurately locate the changed parts of the Service Guide and only to focus on receiving those.

5.5.1.2 Enabling Terminal to determine Service Guide completeness

At each given time, the terminal is proposed a set of SGDD over the Service Guide Announcement Channel, in the form of Transport Objects.

In addition to the construction rules defined in the previous section, in order for the terminal to determine it is receiving a consistent set of the Service Guide, the following signalling in the FLUTE FDT is specified. A new attribute 'FullFDT' is added to the 'FDT-Instance' element of the FDT. This attribute signals a complete description of all the Transport Objects that are currently scheduled for transmission in the corresponding FLUTE session.

The XML syntax of the 'FullFDT' attribute is the following:

```
<attribute name="FullFDT" type="boolean" use="optional" default="false" />
```

When the 'FullFDT' attribute is set to "true" in the FDT instance of a given FLUTE session, this means that all the Transport Objects that are scheduled in the said FLUTE session are described. When the 'FullFDT' attribute is left to false, the terminal cannot make any assumptions about the description. To enable SG-C in the terminal to determine any subset of a Service Guide the SG-D SHALL comply with the following rules:

- In the context of the FLUTE session providing the Service Guide Announcement Channel, the 'FullFDT' attribute SHALL always be set to "true".
- The set of fragments declared by the SGDDs SHALL be exhaustive, i.e., each fragment in the SG has to be declared at least in one SGDD

In addition, should several FDT instances be available at the same time, the terminal SHALL only consider the FDT Instance with the highest value of the FDT Instance ID taking into account the possible wrap around of the FDT Instance ID.

5.5.1.3 Terminal behaviour determining the Service Guide completeness (Informative)

The following state diagram Figure 4 below defines the deduction algorithm for Service Guide completeness when FLUTE is used for SGDD delivery. In principle, the exhaustive listing of fragments makes it possible to determine precisely the fragments that are needed and their location in the Service Guide delivery.

Explanation of variables used in the algorithm:

L_D Set of SGDDs representing a subset of Service Guide.

L_P Previously stored set of SGDDs.

L_R Set of SGDDs to be received.

F_R Set of fragments to be received

F_D Set of fragments to be deleted

C_R Set of SGDUs delivering the fragments in F_R

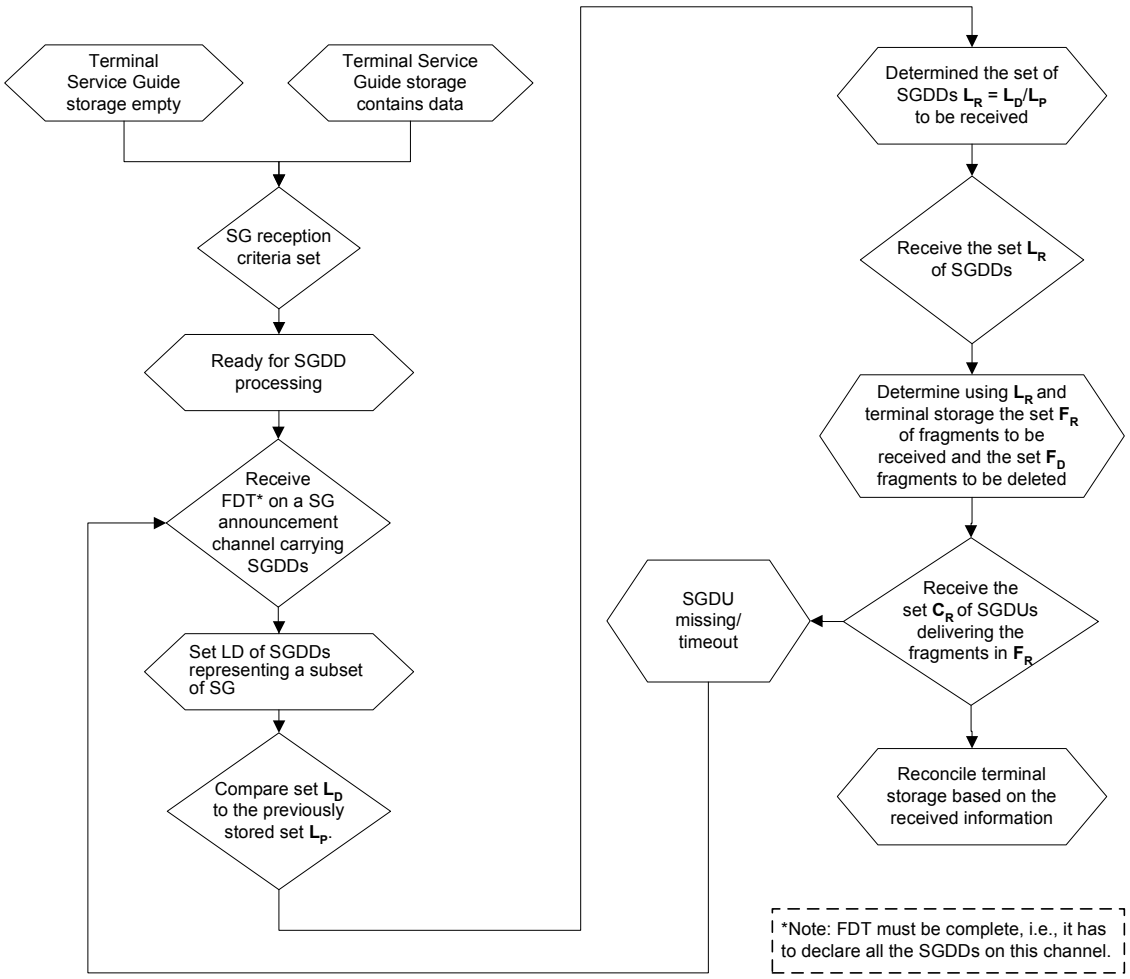


Figure 7: Algorithm for determining a subset of Service Guide

5.5.2 Service Guide update and management using the Interaction Channel

The Service Guide update and management over the Interaction Channel is enabled in two ways: In a terminal-based way and in a system-based way. The terminal-based way and system-based way can be applied mutually complementing each other.

In the terminal-based way the terminal keeps the state of version and validity of acquired Service Guide fragments and Service Guide Delivery Descriptors. Based on that information and the information available via Service Guide Delivery Descriptors possibly made available over Broadcast Channel the terminal detects the changes and reacquires the necessary fragments. The terminal can also poll for changes by trying to reacquire the already acquired Service Guide fragments and/or SGDDs over the Interaction Channel and deduce the changes that way. The delivery of Service Guide over Interactive Channel is normatively specified in section 5.4.3.

In the system-based way the terminal requests the system to keep the terminal updated upon changes on the requested Service Guide fragments and/or on Service Guide Delivery Descriptors. If the terminal supports OMA PUSH the terminal SHALL support the system-based way of update and management using the Interaction Channel as follows:

- Upon terminal requesting Service Guide Delivery Descriptors or Service Guide fragments through means as specified in section 5.4.3., the 'message-body' of HTTP/1.1 request SHALL be suffixed with string "&keep-updated=true" meaning that terminal requests the system to keep the terminal updated on changes to requested Service Guide Delivery Descriptors or Service Guide fragments.

- Upon changes or updates to the Service Guide fragments and/or Service Guide Delivery Descriptors for which the terminal has requested system to keep the terminal updated the following happens. The system SHOULD send the updated Service Guide fragments and/or Service Guide Delivery Descriptors to the terminal using OMA PUSH.

- For Service Guide fragments, the MIME type "application/vnd.oma.bcast.sgdu" SHALL be used to identify that the PUSH message carries an SGDU containing the fragments.
- For Service Guide Delivery Descriptor, the MIME type "application/vnd.oma.bcast.sgdd+xml" SHALL be used to identify that the PUSH message carries an SGDD. Each SGDD will be delivered as a separate PUSH message.

5.5.3 Service Guide Update and Management Cases for Hybrid Broadcast/Interactive Scenario (Informative)

When Service Guide is delivered both over Broadcast Channel and over Interactive Channel, three cases for Service Guide update and management can be envisioned:

- The SG consists of two parts, one being distributed over broadcast (typically a "basic" SG consisting of SGDDs and possibly "basic" fragments, which may for example declare the available services), and one being distributed over interaction channel (the "supplementary" fragments, describing e.g. programs and further details of the services). In this case, the broadcasted parts are updated and managed as described in previous sections. If a "supplementary" fragment changes, the version of the "basic" fragment(s) that are used to retrieve the "supplementary" fragments is increased. If a terminal observes that a "basic" fragment changes, it assumes that the corresponding "supplementary" fragments may have changed. The Terminal can in this case re-acquire those fragments.
- The SG is completely distributed over the interaction channel only. In this case, the SGDDs may contain information announcing notifications (see section 5.4.1.2.5), and the BCAST notification function (see section 5.14 of [BCAST10-Services]) can be used to announce changed SGDDs and.or fragments from server to Terminal. Alternatively, the Terminal can request SGDDs and/or fragments interactively, and determine changes by comparison with the stored SG information.
- The SG is completely broadcasted, and the same version of the SG is also alternatively available over the interaction channel. In this case, the Terminal can use the broadcasted SG to detect changes or updates in the SG, as described in previous sections. Alternatively, the Terminal can request SGDDs and/or fragments interactively, and determine changes by comparison with the stored SG information. Alternatively, the BCAST notification function (see section 5.14 of [BCAST10-Services]) can be used to announce changed SGDDs and.or fragments from server to Terminal.

5.6 Service Guide Backend Interfaces

The following sections specify interfaces between logical BCAST "backend" entities. The specification is applicable if the interfaces are exposed in a BCAST implementation. If a BCAST implementation does not expose the interfaces, i.e. they are implementation internal, they may be realized using protocols and methods not specified here. If a BCAST implementation does expose the interfaces, the network SHALL support the Service Guide Backend Interfaces syntax as defined by XML Schema in [BCAST10-Schema-sg-be].

5.6.1 Service Guide management in the backend

The server side clients of the BCAST represented by the logical entity 'Content Creation', CC, are regarded as providing information about services and/or individual pieces of contents for the BCAST system, which are in turn supplied to the users consuming the services. This exchange of information is performed over the interface SG-1 between the 'Service Guide Content Creation Source' entity, SGCCS, of the CC and the 'Service Guide Application Source' entity, SGAS, of the BSA. The consumers of the services get this information in the form of the Service Guide, SG, via the head-end interfaces SG-5 and SG-6. These interfaces are not provided directly by SGAS, but instead by the 'Service Guide Distribution' entity, SG-D, residing in the BSD/A. The SGAS supplies the information it receives from CC first to the 'Service Guide Generation' entity, SG-G, also residing in BSD/A, over the interface SG-2. The SG-D then receives the information from the SG-G via an internal interface in BSD/A. In addition to receiving information from SGAS, the SG-G also aggregates Service Provisioning information provided by the 'Service Guide Subscription Source' entity, SGSS, of the BSM. This exchange is performed over the interface SG-4.

This arrangement makes it possible for the BSD/A

- to act as a central control point of the program/content control on behalf of the CC, BSA and BSM by collecting information from BSA, BSM and internally from the different entities in BSD/A,

- to act only as an aggregator of the program/content specific portion of the Service Guide in the form of Service Guide fragments to be delivered for the terminals through head-end interfaces SG-5 and SG-6 whereas the BSA is given the control of creating the content-specific Service Guide fragments on behalf of the Content Creation entities or
- to act in some combined mode of the two.

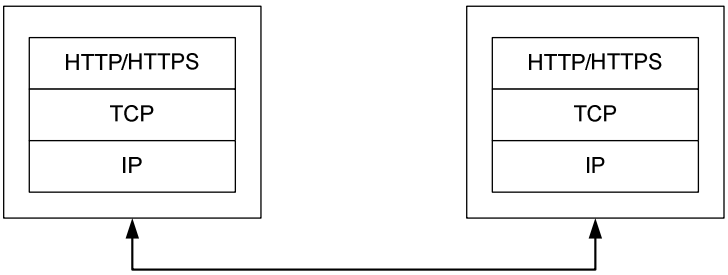
The arrangements in practise between the BSA, BSM and BSD/A with respect to Service Guide management are, however, out of the scope of this specification.

5.6.2 Service Guide contents exchange

This specification defines only the interface SG-4 while leaving interfaces SG-1 and SG-2 undefined.

5.6.2.1 Protocol Stacks

The following protocol stack SHALL be used for exchanging messages between the backend SG entities. HTTP or HTTPS that SHALL be based on SSL 3.0 [SSL30] and TLS 1.0 [RFC2246] over TCP/IP SHALL be used for the delivery of messages.



Messages to and from CC, BSA, BSD/A or BSM are transported using HTTP by placing both the requests and the responses addressed to CC, BSA, BSD/A or BSM into the payload of the HTTP messages. The requests SHOULD be transported using HTTP POST and the responses SHOULD be transported using the HTTP responses corresponding to the HTTP POST requests. The syntax for the requests SHOULD be as follows:

- `POST <host>/oma/bcast1.0/sg HTTP/1.1\r\n<request>`

where the <host> denotes the part of the URI representing the address of the host and the <request> denotes the XML element providing the request parameters.

Both the HTTP POST message and the corresponding HTTP response MAY also contain the following HTTP header fields:

- 'Content-Length',
- 'Content-Type' which if used SHALL be set to "text/xml" and
- 'Host' in case the 'Request-URI' is not in the absolute form specified in [RFC 2616].

5.6.2.2 Service Guide content delivery

This section specifies the service guide delivery message from SGSS to SG-G via SG-4 interface.

5.6.2.2.1 Delivery Message

The following is the delivery message sent from SGSS to SG-G over the interface SG-4. In order to reflect any possible grouping of SG in the head-end interfaces SG-5 and SG-6, SGSS can inform SG-D via SG-G of the grouping by associating each 'SGData' element with one or more 'BSMSelector' elements and/or 'bsmSelectorID' attributes in the 'SGDelivery' request.

Name	Type	Category	Cardinality	Description	Data Type
SGDelivery	E			Specifies the delivery message of Service Guide	

				data over interface SG-4which is used for generating Service Guide in SG-G. Contains the following elements: BSMSelector BSMSelectorID SGData PrivateExt	
BSMSelector	E1	M	0..N	This element provides the details on the visibility of the enclosed 'SGData'. All the 'BSMSelectorID' values used in the request SHALL have one and only one of these 'BSMSelector' elements instantiated with matching identifier. Element 'BSMSelector' is specified in section 5.4.1.5.2.	complexType
BSMSelectorID	E1	M	0..N	This element represents constraints on the visibility of the all enclosed 'SGData' elements. This identifier corresponds to the 'id' attribute of the 'BSMSelector' element. See the corresponding description in the 'SGDD' for more details.	anyURI
SGData	E1	M	1..N	Contains source information to be included into the Service Guide. It is RECOMMENDED that the information is delivered in the form of BCAST Service Guide fragments. Contains the following attributes: id transportID version validFrom validTo encoding type Contains the following element: Body	
id	A	M	0..1	Identifier of the data enclosed in element 'Body'. See also the description of the 'id' attribute in the 'SGDeliveryRes' message.	anyURI
transportID	A	O	0..1	Transport identifier of the data enclosed in element 'Body'.	unsignedLong
version	A	M	1	Version of the data enclosed in element 'Body'.	unsignedInt
validFrom	A	M	1	Start time of the validity of the data enclosed in element 'Body'.	unsignedInt
validTo	A	M	1	End time of the validity of the data enclosed in element 'Body'.	unsignedInt
encoding	A	M	1	Fragment encoding type of the data enclosed in element 'Body'. 0 – XML encoded OMA BCAST Service Guide fragment	unsignedByte

				1 – SDP 2 – MBMS User Service Bundle Description (MBMS-USBD) as specified in [3GPP TS 26.346] section 5.2.2. It may contain one or more SDP descriptions. 3 – AssociatedDeliveryProcedure for File and Stream Distribution as specified in [BCAST10-Distribution] section 5.3.4 4 –127 Reserved for future use 128-255 Reserved for proprietary use	
type	A	M	0..1	Fragment encoding type of the data enclosed in element 'Body'. This SHALL be present and set to one of the values listed below f the 'Body' contains a Service Guide XML fragment and the value of 'encoding' is set to '0'. 5 – PurchaseItem Fragment 6 – PurchaseData Fragment 7– PurchaseChannel Fragment	unsignedByte
Body	E2	M	1	Contains the delivered Service Guide data. The value SHALL be an instance of 'PurchaseItem', 'PurchaseData' or 'PurchaseChannel' element as specified in 5.1.2.6, 5.1.2.7 and 5.1.2.8 of [BCAST10-SG].	complexType
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.6.2.2.2 Response Message

The following is the response message for 'SGDelivery' request to be used over interface SG-4.

Name	Type	Category	Cardinality	Description	Data Type
SGDelivery Res	E			Specifies the response message to be used in the response to 'SGDelivery' HTTP request. Contains the following elements: SGDataResult PrivateExt	
SGDataResult	E1	M	0..N	Represents the status for each of the 'SGData' elements of the corresponding HTTP request. If there are more than one 'SGData' element present in the HTTP request, these elements SHALL correspond to the 'SGData' elements of the HTTP request in the same order as in the HTTP request. Contains the following attributes: id statusCode	
id	A	M	1	Identifier of the Service Guide data contained in the 'SGData' element of the HTTP request. The assignment of this identifier is performed as	anyURI

				follows: CASE 1: The value of the 'id' attribute in the 'SGData' element of the 'SGDelivery' HTTP request equals 'oma:bcast:unknown'. In this case the identifier of the fragment SHALL be assigned by the issuer of the response message and the value of the identifier SHALL be set as the value of the 'id' in this response. CASE 2: The value of the 'id' attribute in the 'SGData' element of the HTTP request does not equal 'oma:bcast:unknown'. In this case the identifier of the fragment SHALL be interpreted as assigned by the issuer of the 'SGDelivery' HTTP request and the issuer of the response SHALL simply copy the value of the 'id' attribute of the request into the 'id' attribute of the response. The issuer of the 'SGDelivery' HTTP request SHALL use this returned identifier in any future 'SGDelivery' HTTP requests representing updates on the information in the 'SGData' element in question.	
statusCode	A	M	1	Indicates the outcome of the processing of the 'SGData' element with matching value of 'id' attribute. The value SHALL be set as specified in [BCAST10-Services] for global status codes.	unsignedByte
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.6.2.3 Service Protection Information Requests

In order to be able to fill in information about possible service and content protection into 'Service' or 'Content' fragments, the SG-G of BSD/A needs to be able to request the corresponding parameters from the SGSS of BSM representing in turn access to the 'SP Management' of BSM. This request from SG-G to SGSS over SG-4 SHALL be an HTTP POST having an instantiation of the following 'SGSPServiceInfo' element as the payload.

Name	Type	Category	Cardinality	Description	Data Type
SGSPServiceInfo	E			Specifies the message for SG-G to request the value of the 'baseCID' and 'ProtectionKeyID' to be used in a 'Service' or 'Content' fragment and to query if a particular service should be signalled as protected or not from the SGSS. Contains the following attributes: globalServiceID globalContentID Contains the following element: PrivateExt	
globalServiceID	A	M	0..1	Globally unique identifier of the service in question. This is the value of the element 'GlobalServiceID' in the 'Service' fragment representing the service in question.	anyURI

globalContentID	A	M	0..1	Globally unique content identifier of the piece of content in question. This is the value of the element 'GlobalContentID' in the 'Content' fragment representing the piece of content in question.	anyURI
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

The following is the response message for the 'SGSPServiceInfo' to be used over the interface SG-4. The response from SGSS of BSM to SG-G of BSD/A SHALL be an instantiation of the following 'SGSPServiceInfoRes' message placed into the payload of the corresponding HTTP response.

Name	Type	Category	Cardinality	Description	Data Type
SGSPServiceInfoRes	E			Specifies the response message for a 'SGSPServiceInfo'. Contains the following attributes: statusCode serviceContentProtection baseCID Contains the following elements: ProtectionKeyID PrivateExt	
statusCode	A	M	1	Specifies the result of the 'SGSPServiceInfo' this response corresponds to using the Status Code as specified in [BCAST10-Services].	unsignedByte
serviceContentProtection	A	M	0..1	Specifies if the service is encrypted (true) or not (false). This attribute SHALL not be instantiated when the corresponding request addresses 'Content' fragment using the 'GlobalContentID' element.	boolean
baseCID	A	M	0..1	The value to be filled in by the BSM as the value of the 'baseCID' attribute of the corresponding 'Service' or 'Content' fragment. If this attribute is omitted in the response, the BSA SHALL not instantiate the attribute 'baseCID' in the corresponding 'Service' or 'Content' fragment.	string
ProtectionKeyID	E1	O	0..N	The value to be filled in by the BSM as the value of the 'ProtectionKeyID' element of the corresponding 'Service' fragment. If this element is omitted in the response, the BSM SHALL not instantiate the element 'ProtectionKeyID' in the corresponding 'Service' fragment. This attribute SHALL not be instantiated when the corresponding request addresses 'Content' fragment using the 'GlobalContentID' element.	base64Binary

				Contains the following attribute: type	
type	A	M	1	Type of ProtectionKeyID: For the possible values of this atttribute, see 'Service' fragment specififed in section 5.1.2.1.	unsignedByte
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

In order to be able to fill in information about possible service and content protection in 'Access' fragments, the SG-G of BSD/A needs to be able to request the corresponding parameters from the SGSS of BSM representing in turn access to the 'SP Management' of BSM. This request from SG-G to SGSS over SG-4 SHALL be an HTTP POST having an instantiation of the following 'SGSPAccessInfo' element as the payload.

Name	Type	Category	Cardinality	Description	Data Type
SGSPAccessInfo	E			Specifies the message for SG-G to request the values of the 'KeyManagementSystem' and 'EncryptionType' elements to be used in an 'Access' fragment from the SGSS. Contains the following attribute: globalServiceID Contains the following element: PrivateExt	
globalServiceID	A	M	1	Globally unique identifier of the service providing indirect identification of the access(es) in question. This is the value of the element 'GlobalServiceID' in the 'Service' fragment into the which 'Access' fragment in question is associated to.	anyURI
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

The following is the response message for the 'SGSPAccessInfo' to be used over the interface SG-4. The response from SGSS of BSM to SG-G of BSD/A SHALL be an instantiation of the following 'SGSPAccessInfoRes' message placed into the payload of the corresponding HTTP response.

Name	Type	Category	Cardinality	Description	Data Type
SGSPAccessInfoRes	E	M		Specifies the response message for a 'SGSPAccessInfo'. Contains the following attribute: statusCode Contains the following elements: KeyManagementSystem	

				EncryptionType PrivateExt	
statusCode	A	M	1	Specifies the result of the 'SGSPAccessInfo' this response corresponds to using the Status Code as specified in [BCAST10-Services].	unsignedByte
KeyManagementSystem	E1	M	0..1	The value to be filled in by the BSD/A as the value of the 'KeyManagementSystem' element of the corresponding 'Access' fragment. If this element is omitted in the response, the BSA SHALL not instantiate the element 'KeyManagementSystem' in the corresponding 'Access' fragment. The element 'KeyManagementSystem' is specified in section 5.1.2.4.	complexType
EncryptionType	E1	M	0..N	The value to be filled in by the BSA as the value of the 'EncryptionType' element of the corresponding 'Access' fragment. If this element is omitted in the response, the BSA SHALL not instantiate the element 'EncryptionType' in the corresponding 'Access' fragment.	unsignedByte
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.6.2.4 Service Provisioning Information Request

In order for SG-G of BSD/A to be able to provide provisioning fragments over interfaces SG-5 and SG-6, the SG-G needs to be able to request these from the SGSS of BSM. This request from SG-G to SGSS over SG-4 SHALL be an HTTP POST having an instantiation of the following 'SGProvisioningInfo' element as the payload.

Name	Type	Category	Cardinality	Description	Data Type
SGProvisioningInfo	E			Specifies the message for SG-G to request provisioning fragments from the SGSS. Contains the following elements: GlobalServiceID GlobalContentID PrivateExt If both elements are omitted, the SGSS SHALL interpret this message requesting for 'PurchaseChannel' fragments.	
GlobalServiceID	E1	M	0..1	Globally unique identifier of the service for which the SGAS request provisioning fragments.	anyURI
GlobalContentID	E1	M	0..1	Globally unique identifier of the piece of content for which the SGAS request provisioning fragments.	anyURI
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These	

				elements may further contain sub-elements or attributes.	

The following is the response message for the 'SGProvisioningInfo' to be used over the interface SG-4. The response from SGSS of BSM to SG-G of BSD/A SHALL be an instantiation of the following 'SGProvisioningInfoRes' message placed into the payload of the corresponding HTTP response.

Name	Type	Category	Cardinality	Description	Data Type
SGProvisioningInfoRes	E	M		Specifies the response message for a 'SGProvisioningInfoRes'. Contains the following attribute: statusCode Contains the following elements: ProvisioningFragment PrivateExt	
statusCode	A	M	1	Specifies the result of the 'SGProvisioningInfo' this response corresponds to using the Status Code as specified in [BCAST10-Services].	unsignedByte
Provisioning Fragment	E1	M	0..N	This element contains the SG data in question as a standard BCAST provisioning fragment.	complexType
PrivateExt	E1	O	0..1	An element serving as a container for proprietary or application-specific extensions.	
<proprietary elements>	E2	O	0..N	Proprietary or application-specific elements that are not defined in this specification. These elements may further contain sub-elements or attributes.	

5.7 Use of PreviewData to Support Service Switching, Browsing and Preview

Within the Service Guide, any or all of the Service, Content, Access, Schedule and 'PurchaseData' fragments MAY be associated with 'PreviewData' fragments. The 'PreviewData' fragment can include simple text, binary images, references to images, video clips and audio clips, or the combination of the aforementioned, or reference another service which could be a low bit rate version for the main service. The references to these media components of the preview data can be provided with remote URIs (for example, HTTP URL, RTSP URL).

The following uses are specified for 'PreviewData' fragment:

(1) Service-by-Service Switching: rendering the preview data associated with 'PreviewData' fragment

- during the waiting time while terminal discovers of entry point to, and actually joins a service or content item, or
- during the delay associated to user-initiated switching from one service/content item to another.

(2) Service Guide Browsing: using the preview data associated with 'PreviewData' fragment as the representative item (eg. icon, logo) of a service, content, or purchase package.

(3) Service Preview: presenting the preview data associated with 'PreviewData' fragment to users when the users want to briefly preview the service, content or purchase package;

(4) Barker: presenting the preview data associated with 'PreviewData' fragment to users in place of a selected service or content to which the user does not have rights to access or is not subscribed.

(5) Terms of Use: presenting the preview data associated with 'PreviewData' fragment as a representation of Terms of Use. This use of 'PreviewData' is allowed in a Service Guide fragment only when 'TermsOfUse' element is present in that fragment.

(6) Alternative to blackout: presenting the preview data associated with 'PreviewData' fragment to users when the terminal has determined that the requested service is under blackout, as specified in [BCAST10-SvcCntProtection] section 7.1.

When a reference to 'PreviewData' fragment is made from another fragment for the uses (1), (2), (3), (4) or (6) above, the following applies:

- The referring fragment SHALL assign E1 element 'PreviewDataReference' with the fragment id of the referred 'PreviewDataFragment'.
- The attribute 'usage' of the 'PreviewDataReference' in the referring fragment SHALL be assigned with value '1', '2', '3', '4' or '5'.

When a reference to 'PreviewData' fragment is made from another fragment for the use (5) above, the following applies:

- The referring fragment SHALL assign E2 element 'PreviewDataIDRef' with the fragment id of the referred 'PreviewDataFragment'.
- The 'PreviewDataIDRef' in the referring fragment SHALL be a sub-element of 'TermsOfUse' E1 element.

It is possible that there are more than one 'PreviewData' fragments associated with the same Service, Content, Access, Schedule or PurchaseData fragment. In this case multiple instances of the 'PreviewDataReference' element will be present in the associated fragment; however the values of the 'usage' attributes of these 'PreviewDataReference' instances SHALL be mutually exclusive.

The following Table 4 summarizes the allowed uses of 'PreviewData' fragment when referenced by other Service Guide fragments for various usages. In this table, "X" means: the fragment that appears on the respective row can reference 'PreviewData' fragment for the usage stated on the respective column. The terminal SHALL support each of the identified uses of 'PreviewData' fragment, but MAY NOT support those unidentified uses of 'PreviewData' fragment. For instance, the Service fragment can reference the 'PreviewData' fragment as the representative item of the associated service for "Service Guide Browsing", however while the 'PreviewData' fragment is associated with Service fragment with use "Service-by-Service Switching", there is no guarantee that a BCAST 1.0 terminal supports such use.

	Service-by-Service Switching	**Service Guide Browsing**	**Service Preview**	**Barker**	**Terms of Use**	**Blackout**
Service fragment		X (as representative item of the service)	X (as preview of the service)	X (when no rights to view the selected service)	X (present only within element 'TermsOfUse')	
Content fragment		X (as representative item of the content)	X (as preview of the content)	X (when no rights to view the selected content)	X (present only within element 'TermsOfUse')	
Schedule fragment		X (as representative item of the schedule)	X (as preview of the schedule)		X (present only within element 'TermsOfUse')	
Access fragment	X					X
PurchaseData		X	X		X	

fragment		(as representative item of the purchase data)	(as preview of the purchase data)		(present only within element 'TermsOfUse')	
InteractivityData fragment					X (present only within element 'TermsOfUse')	

Table 4: Uses of 'PreviewData' fragment when referenced by other Service Guide fragments

The details of the above usages of the preview data are further explained in the following sub-sections.

5.7.1 Preview Data for Service-by-Service Switching

When switching services, the terminal can render the preview data declared by 'PreviewData' fragment during the delay associated with terminal discovering of the entry point and actually joining to receive a service or content, or between user-initiated switching from one service or content to another. Which part or what media components of the preview data are used in rendering and with what layout is specific to each implementation. For example, a lower-end terminal could render just the descriptive text while a higher-end terminal could render the associated video clip.

This feature can be enabled by declaring the identifier of the 'PreviewData' fragment in the associated 'Access' fragment, and indicating the usage value as "1" (Service-by-Service Switching). Upon accessing content or a service, which has a related Access fragment with a reference to a 'PreviewData' fragment, the media components declared in this 'PreviewData' fragment SHOULD be received and rendered by the terminal, if available, before and until the main service appears.

The informative diagram Figure 8 below illustrates an example case how preview data supports Service-by-Service switching. In the diagram, the user first switches from Service X to Service Y (step 1a) and the terminal correspondingly switches from the Access related with Service X to the Access related with Service Y (step 1b). Consequently, the preview data referenced from Access Fragment related with Service Y is rendered (step 2). Finally, after stream initialization delay, content of Service Y is rendered (Step 3). At this point of time the preview data is removed from the screen. The following rendering for Service-by-Service switching is similar.

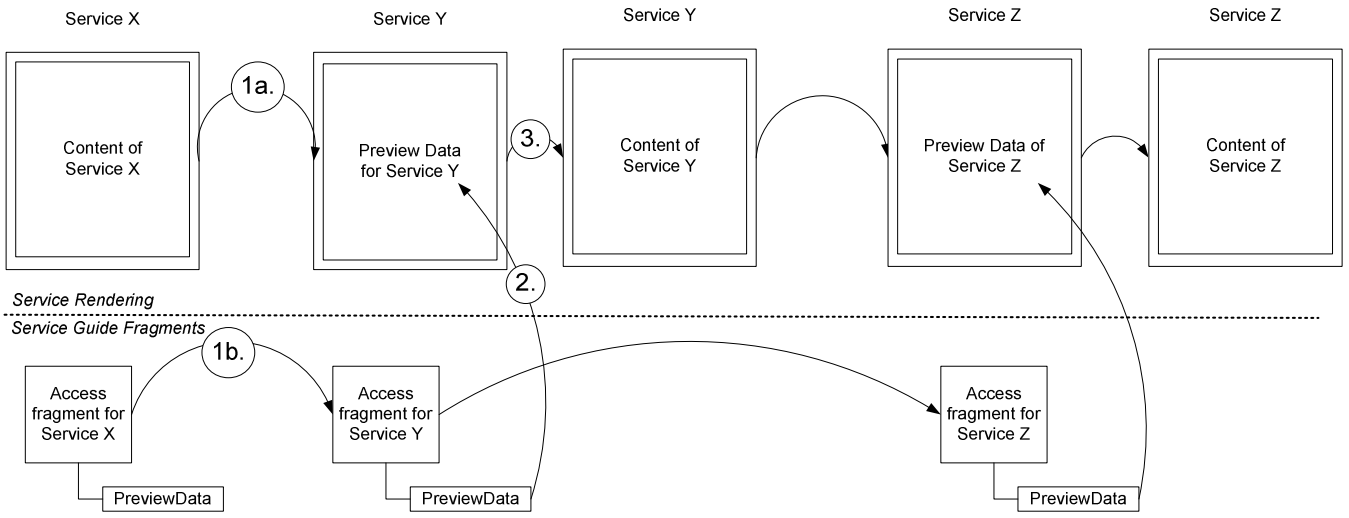


Figure 8: Example case how preview data supports service-by-service switching

5.7.2 Preview Data for Service Guide Browsing

When browsing services, contents or purchase packages as a list, a tree or similar structure on the terminal, multiple services, contents, services or contents within a specific time slot, or purchase packages will be visible at the same time. In such case the terminal can render the associated preview data as the representative item (eg. icon, logo) of each service, content,

schedule or purchase package. Which part or what media components of the preview data are used in rendering and with what layout is specific to each implementation. For example, a lower-end terminal could render just a small icon next to the service or content title while a higher-end terminal could render a simplified version of the actual service or content in a small size next to the service or content title and description.

This feature can be enabled by declaring the identifier of the 'PreviewData' fragment in the associated Service, Content, Schedule or PurchaseData fragments, and indicating the usage value as "2" (Service Guide Browsing).

The informative diagrams Figure 9 and Figure 10 below illustrate the support for Service Guide browsing with schedule view and purchase view. In Figure 9, the Service Guide browser displays multiple services and content in tree structure according to the scheduling. In this example the preview data of Service X, Content A, Content B, Content C and Schedule AM are shown as small icons next to the tile of the associated service, content and schedule. Similarly, in Figure 6.2 the Service Guide browser displays multiple services according to the purchase packaging. In this example, the two purchase packages corresponding to purchase data A and B are associated with the same purchase item X with different pricing scheme a and b respectively, hence the services belonging to purchase data A and B are identical (i.e. service i and service j), but the pricing scheme and other information of these two purchase data are different.

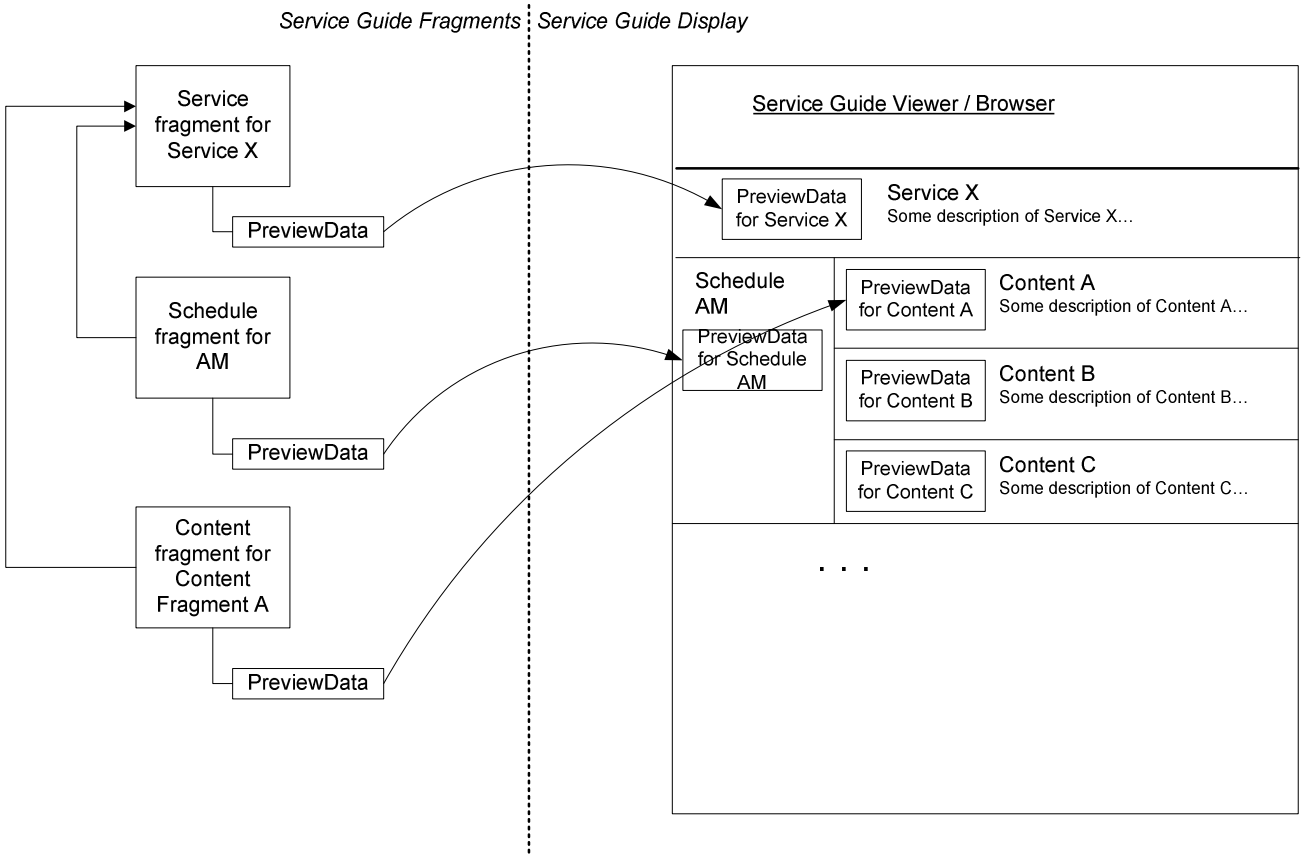


Figure 9: Example case how preview data supports Service Guide browsing with schedule view

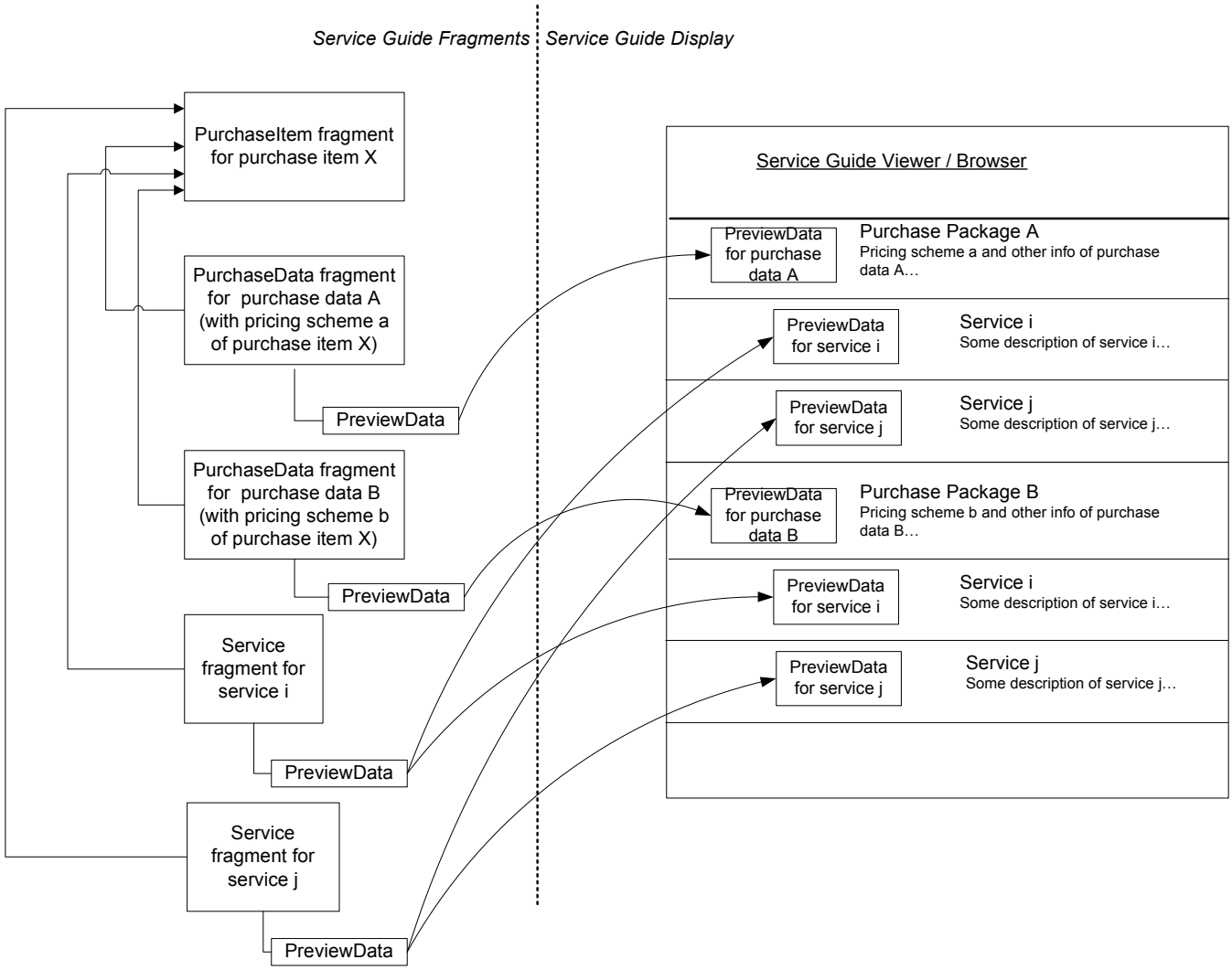


Figure 10: Example case how preview data supports Service Guide browsing with purchase view

5.7.3 Preview Data for Service Preview

When viewing the detailed information of one specific service, content, schedule or purchase package on the terminal, the terminal can render the associated preview data to the users as an advance showing part (eg. trailer, posters) of the associated services, content, schedule or purchase package. Which part or what media components of the preview data are used in rendering and with what layout is specific to each implementation. For example, a lower-end terminal could render just a big poster image(s) while a higher-end terminal could render an A/V trailer in a big size or even full screen as the service preview.

This feature can be enabled by declaring the identifier of the 'PreviewData' fragment in the associated Service, Content, Schedule or PurchaseData fragments, and indicating the usage value as "3" (Service Preview).

The informative diagram Figure 11 below illustrates the support for service preview.

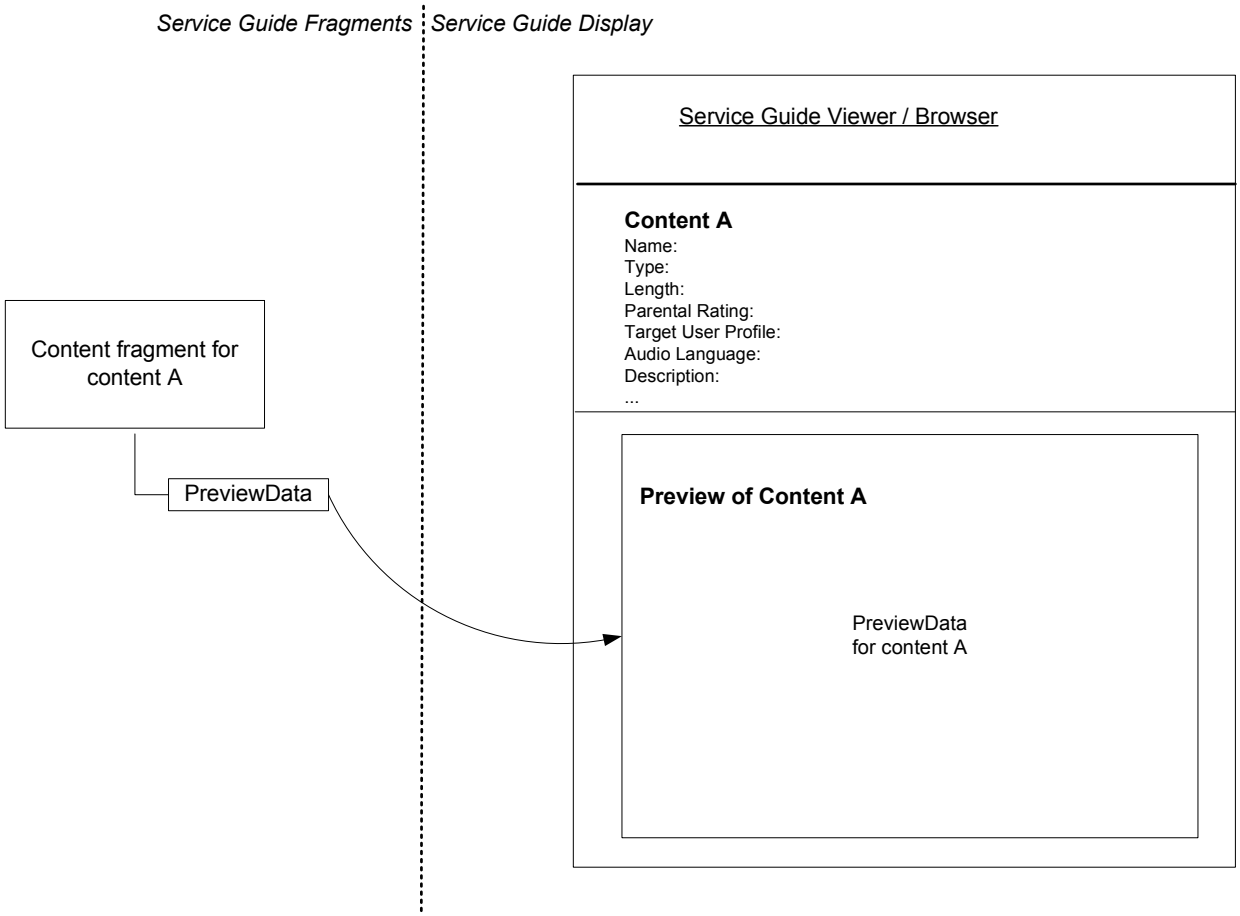


Figure 11: Example case how preview data supports service preview

5.7.4 Preview Data for Barker

The BCAST services can be provided with or without Service or Content Protection. Even when the service is protected and a particular user does not have rights to access the service, the preview data of the service can be visible in the Service Guide. When browsing the services or contents within the Service Guide the user might want to select the service or content which the user does not have rights to access or has not subscribed to, the preview data can be rendered in place of the actual service or content to give the user a preview on the selected service or content, or inform the user he/she does not has the rights to access or has not subscribed to the selected service or content.

This feature can be enabled by declaring the identifier of the 'PreviewData' fragment in the associated Service or Content fragments and indicating the usage value as "4" (Barker).

The informative diagram Figure 12 below illustrates the support for barker.

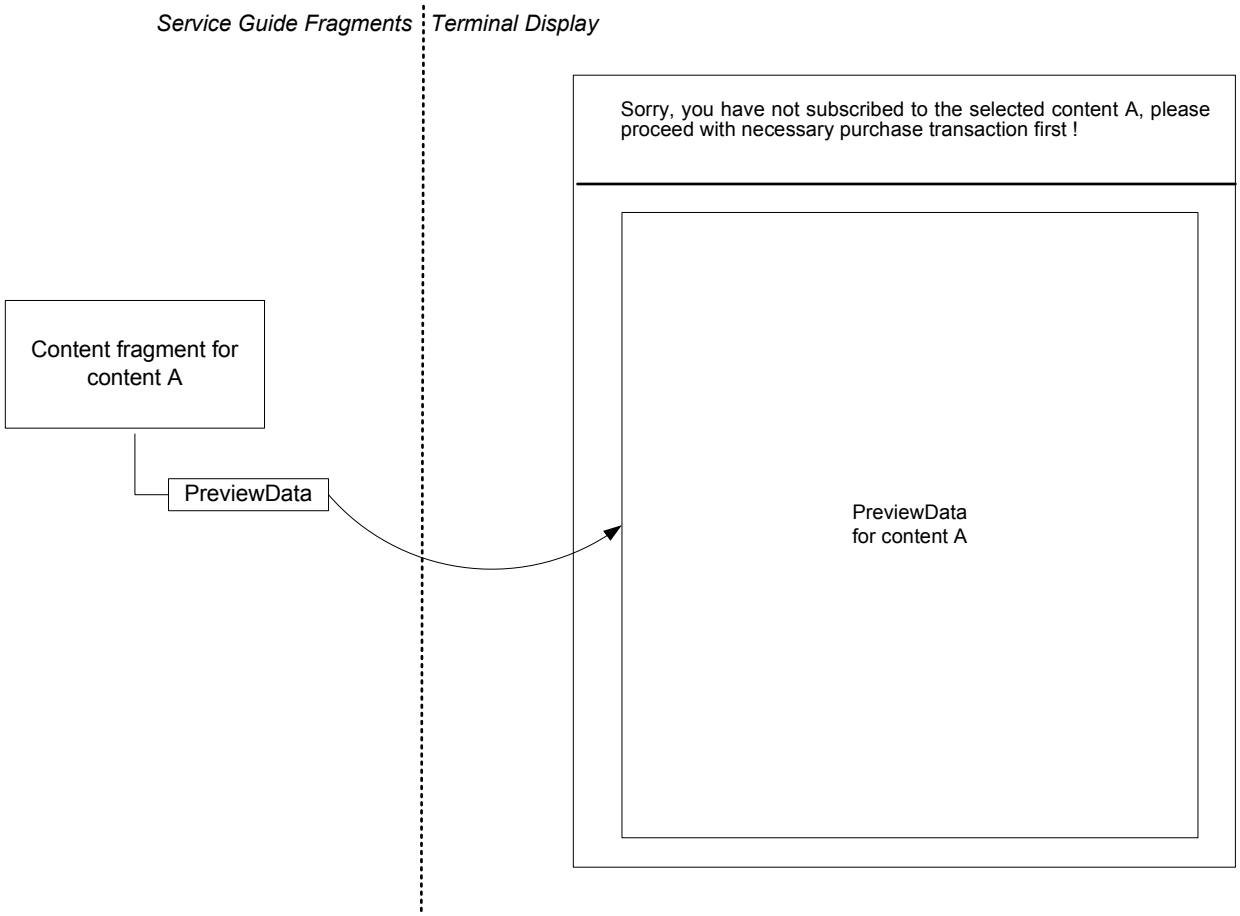


Figure 12: Example case how preview data supports barker

5.7.5 Preview Data as an alternative to blackout

Upon selection of a BCAST service by the end-user, the terminal fetches the corresponding Access fragment and performs the necessary steps to render the service. When the service is protected, the terminal may receive an STKM carrying an Access Criteria Descriptor enforcing a blackout in the current area of reception. In that case, the terminal has the possibility to check the Access fragment for a reference to a 'PreviewData' fragment with a 'usage' value of 5. If such 'PreviewData' fragment exists, it contains alternative payload the terminal can render instead of the blacked-out service. The 'PreviewData' fragment could also point to an Access fragment. In such a case the terminal can use that Access fragment to tune to an alternate service. The figure below illustrates these possibilities.

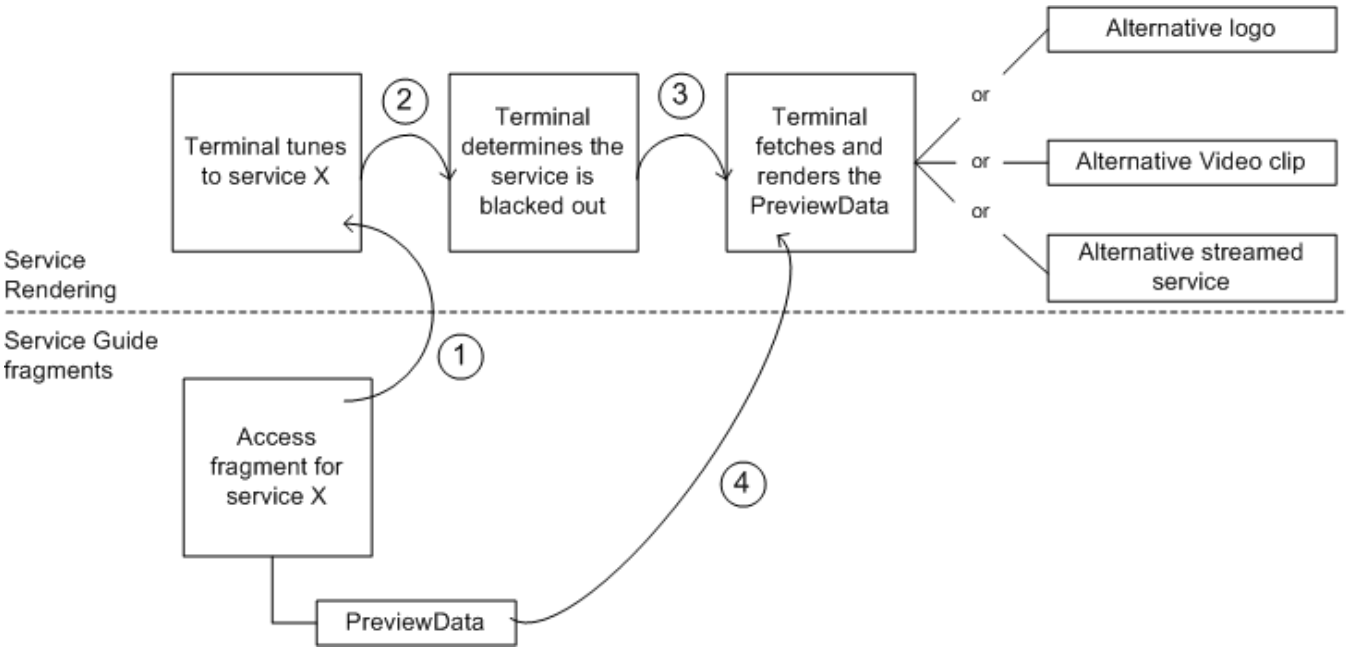


Figure 133: Example case how preview data supports alternative to blackout

5.8 Associating accesses with services and content

There can be different choices for the end users to consume services. These choices are represented using the 'Access' fragments associated to 'Service' fragments and (indirectly via 'Schedule' fragments) 'Content' fragments, the latter two representing services and individual pieces of the services respectively. The mechanisms for associating 'Service' and 'Content' fragments with 'Access' fragments, however, have different characteristics. Whereas a service is not always bound by time, a piece of content is. This is reflected by content having more specialized association to the access details than the service. The following sections define how the network instantiates Service Guide fragments in practice for signaling access parameters and how the terminal is expected to interpret them. Note that the distribution and presentation semantics for "Cachecast" services are different, and described separately in section 5.8.6.

5.8.1 Multiple accesses

Each 'Access' fragment represents different aspect of consumption of a service, or a content item belonging to that service, that 'Access' fragment is associated with. The network can give the terminal and to the end user different options for consuming services. In such a case the corresponding 'Service' or 'Content' fragment could be associated with more than one 'Access' fragment. The different options could be meant for the terminal to make the choice on behalf of the end user or by the end user himself/herself. In either case there needs to be a common understanding between the network and the terminal on how the choice is meant to be taken.

5.8.1.1 Choices of access

When the network provides different means of accessing a service or content items of that service the network also needs to provide a meaning for each choice. Depending on the criterion for which each choice is based, the terminals first decides whether the selection is meant for terminal or the end user to make. In case the network provides multiple 'Access' fragments associated with 'Service' or 'Content' fragment, regarding those information contained in the 'Access' fragment meant for the terminal to use, transparent to the end user:

- the network SHALL ensure that these 'Access' fragments differ in value with respect to at least one of the following elements:
    - 'AccessType'(e.g. 'BDSType')
    - 'KeyManagementSystem'
    - 'EncryptionType'

  - 'TerminalCapabilityRequirement'
  - 'BandwidthRequirement'
  - 'ServiceClass'
- the terminal SHALL choose an 'Access' fragment that matches the best the terminal capabilities and/or settings (e.g. user settings). How the terminal actually makes the choice is out of scope of this specification.

On the other hand, if in turn the choice is meant to be taken by the end user, the network SHALL ensure that the 'audioLanguageIDRef' or the 'textLanguageIDRef ' attributes of the 'ServiceReference' or 'ContentReference' elements in the 'Schedule' fragment are different. For the details on the latter two see section 7 Multi-language Support.

The network SHALL declare the time interval when an 'Access' fragment applies as follows:

- Using the "t=" field of the SDP formatted Session Description instantiated in either the 'Access' or' 'SessionDescription' fragment. In case the field has the value" t=0 0" it is considered to be an unbounded live stream.
- Using the 'startTime' and 'endTime' attributes of the 'PresentationWindow' element in a 'Schedule' fragment that the 'Access' fragment is associated with.
- Should the PresentationWindow be present, it is RECOMMENDED that the time period defined by <start time> and <stop time> in the Session Description encompasses the one defined by the PresentationWindow.

The terminal SHALL consider the latter first and in case no such 'Schedule' fragment or 'PresentationWindow' is available, the terminal SHALL consider the former.

5.8.2 Association between 'Service' and 'Access' fragment

The network can signal a service to be consumed by direct refererence of one or more 'Access' fragment(s) to the 'Service' fragment representing the service. In such a case

- the terminal SHALL assume that the consumption method represented by the 'Access' fragment is available
  - during the interval constraint by the associated Session Description or
  - any time in case of absence of the constraint.
- In order for the terminal make a distinction between the different access methods the network SHALL take into account the constraints in the section 5.8.1 Multiple accesses and
- furthermore the network SHALL instantiate in the maximum one 'Access' fragment referring to the 'Service' fragment where the 'Access' fragment declares a notification component associated with the service using the element 'NotificationReception'. The network MAY declare several 'Access' fragments referring to the 'Service' fragment given that not more than one 'Access' fragment contains such a notification component. Note: There is no meaningful way for the terminal to choose between two notification services and hence the restriction.

5.8.3 Association between 'Service' and 'Access' fragment through 'Schedule' fragment

In addition to the case described in the previous section, the network can also associate a 'Service' fragment with 'Access' fragments using 'Schedule' fragments. This mechanism is used to enable the concept of multi-language support as discussed in details in section 7. However, the basic intention is to represent a single language-specific choice of consumption to the end user. Therefore, the network SHALL instantiate each of the 'Schedule' fragments in question

- with the 'ServiceReference' element referring to the 'Service' fragment but
- without any instantiations of the element 'ContentReference'
- Without any 'onDemand' attribute set to "true".

In order for the terminal to make a distinction between the different access methods the network SHALL take into account the constraints in section 5.8.1 (Multiple accesses) and furthermore the network SHALL not associate any notification component with the service using the element 'NotificationReception' in the 'Access' fragment in question. If there are

multiple ‘Schedule’ fragments referring to the ‘Service’ fragment, at most one fragment SHALL have the ‘defaultSchedule’ attribute set to “true’.

5.8.4 Association between ‘Content’ and ‘Access’ fragment through ‘Service’ or ‘Schedule’ fragment

The ‘Content’ fragment provides means for the network to declare for the end user that a particular service is constructed of individual content parts. For example, traditional broadcast stations often make themselves known to the end users as a service but provide their services as individual pieces of content often also called as programs. The consumption of a single piece of content can be made available to the end user using the following four mechanisms:

- using a scheduled transmission over a broadcast access,
- using a scheduled transmission over an interactive access,
- using an end user initiated consumption of the content over the interaction channel (i.e. ‘OnDemand’) or
- using a (previously) downloaded file.

In the first two cases the availability of a single piece of content for the end user to consume is restricted implicitly. The stream for a particular content is available only during a specific interval of time and the network can also control when the terminals are allowed to access content over the interaction channel. The latter case, however, requires the terminal to control when the end user is allowed to consume the downloaded file.

5.8.4.1 Content access through service access

‘Content’ fragments can ‘inherit’ the ‘Access’ fragments that are associated with the ‘Service’ fragment. This is the use case where the network uses the same broadcast access for the service itself as for the individual pieces of content of the service. For this the network associates

- An ‘Access’ fragment directly with a ‘Service’ fragment as specified above in section 5.8.2 or
- An ‘Access’ fragment with a ‘Schedule’ fragment that is associated with the ‘Service’ fragment as specified above in section 5.8.3

In order to enable the terminal to interpret the use case correctly, each of the ‘Content’ fragments in question SHALL contain one and only one instantiation of the ‘ServiceReference’ element referring to the ‘Service’. In order for the terminal to represent these different methods of consumption for the end user to choose from, the terminal needs to comply with the following:

In the first case the terminal SHALL consider an ‘Access’ fragment (that is associated with a service) to be associated with a ‘Content’ fragment if

- the ‘Content’ and ‘Access’ fragment(s) in question refer to the same ‘Service’ fragment. In this case all those text languages or audio languages of content items in the ‘Content’ fragments (see section 7 Multi-language Support) which do not have their own dedicated ‘Schedule’ fragment (as outlines below or in section 5.8.4.2) are assumed to be accessible using the ‘Access’ fragment that is associated with the ‘Service’ fragment (either directly or through a ‘Schedule’ fragment as defined in section 5.8.2 or 5.8.3).
- and there is no ‘Schedule’ fragment associated with the same Content fragment that has the ‘onDemand’ attribute set to ‘true’

In the second case the terminal SHALL consider a ‘Schedule’ fragment (that is associated with a service only) and its associated ‘Access’ fragment to be associated with a ‘Content’ fragment

- If the ‘Content’ and ‘Schedule’ fragment(s) in question refer to the same ‘Service’ fragment, the ‘Content’ fragment refers to one and only one ‘Service’ fragment
- If the ‘Schedule’ fragment contains attribute ‘audioLanguageIDRef’ whose value matches that of the ‘id’ attribute of the ‘AudioLanguage’ element of the ‘Content’ fragment, the schedule and its associated ‘Access’ fragment applies to that particular audio language.

- If the 'Schedule' fragment contains attribute 'textLanguageIDRef' whose value matches that of the 'id' attribute of the 'TextLanguage' element of the 'Content' fragment, the schedule and its associated 'Access' fragment applies to that particular text language.
- There is no 'Schedule' fragment associated with the same Content fragment that has the 'onDemand' attribute set to 'true'.

5.8.4.2 Content specific access via 'Schedule' fragment

There can be cases where it is desired for an individual piece of content of a service to have access methods specific only to the individual piece of content. These access methods are associated with the content using 'Schedule' fragments referring to the 'Content' and 'Access' fragments referring to the 'Schedule' fragments in turn. Similarly as in the case of service access described above in section 5.8.2, the network needs to comply to the following rules in order for the terminals to interpret this use case correctly. In such a case the network SHALL instantiate a 'Schedule' fragment in accordance with the following:

- with the 'PresentationWindow' element declaring the available time period of rendering. This PresentationWindow element SHOULD have the startTime and endTime attribute matching the "t=" field in the SDP formatted Session Description information instantiated by either the 'Access' fragment (referencing this 'Schedule' fragment) or the 'SessionDescription' fragment, with the exception when the time field is set to "t=0 0".
- with the 'ContentReference' element referring to the 'Content' fragment but
- without instantiation of the element 'InteractivityReference'.
- and the 'Content' fragment and the 'Schedule' fragment SHALL refer to the same 'Service' fragment.
- in order for the terminal to make a distinction between the different access methods the network SHALL take into account the constraints in the section 5.8.1 Multiple accesses and
- furthermore the network SHALL not associate any notification component with the service using the element 'NotificationReception' in the 'Access' fragment in question.

The terminal in turn SHALL expect that the consumption method(s) represented by the 'Access' fragment(s) can only be used during the time period restricted by the 'PresentationWindow' element of the 'Schedule' fragment.

Furthermore the network MAY set the 'onDemand' attribute to "true" in the 'Schedule' fragment.

- In case there are one or more 'Schedule' fragments related to a 'Content' fragment that have the 'onDemand' attribute set to 'true', the terminal SHALL expect that that content item is only accessible through those 'Schedule' fragments, i.e. the 'Access' fragment associated with the 'Service' fragment this 'Schedule' fragment and related 'Content' fragment is associated with is not valid for that 'Content' item.
- and i case the 'onDemand' attribute is false or not declared in any of the 'Schedule' fragments referring to the 'Content' fragment, the terminal SHALL expect that the content item is also accessible through the 'Access' fragment associated with the 'Service' fragment as described in section 5.8.4.1.

5.8.4.2.1 PresentationWindow, StartTime and EndTime

When a 'Content' fragment is associated with a 'Schedule' fragment the terminal is provided a pair of time intervals

- one interval declared by the 'StartTime' and 'EndTime' elements of the 'Content' fragment and
- another declared by the 'PresentationWindow' of the 'Schedule' fragment.

The former is intended for direct display to the end user the fixed interval during which the corresponding piece of content can be consumed by the end user. The latter may either be used to signal to the terminal when the specific access method represented by the 'Schedule' fragment and the associated 'Access' fragment is to be used.

In case the interval declared by the 'PresentationWindow' does not cover the entire interval declared by the 'StartTime' and 'EndTime' elements of the 'Content' fragment, the terminal SHALL cover the remaining time using one of the 'Access' fragments associated with the 'Service' fragment the 'Content' fragment is associated with or using one of the 'Access' fragments associated with other 'Content' fragments of the same service. In case no such 'Access' fragment is available there are no expectations defined by this specification for the terminal. If in turn there are multiple such accesses available the terminal SHALL choose as specified in sections 5.8.1.1.

5.8.5 Use of ‘defaultSchedule’ and ‘onDemand’ attributes

The usage of the ‘defaultSchedule’ and ‘onDemand’ attribute defines the access method that the terminal has to take into account when entering the ‘Service’ and the access method that takes precedence.

The ‘defaultSchedule’ attribute indicates the access method that takes precedence in case multiple access methods are defined for a certain content item. If there are multiple ‘Schedule’ fragments referring to the ‘Content’ fragment, at most one fragment MAY have the ‘defaultSchedule’ attribute set to ‘true’. A ‘Schedule’ fragment that has the ‘defaultSchedule’ attribute set to ‘true’ and has a link to the ‘Content’ fragment takes precedence over the ‘Schedule’ fragment that has the ‘defaultSchedule’ attribute set to ‘true’ but only has a link to the ‘Service’ fragment of which that ‘Content’ fragment is associated to. It should be noted that specific terminal (language) settings may override the ‘defaultSchedule’.

The ‘onDemand’ attribute indicates the scheduled transmission of content items that are accessible over the unicast channel where the transmission is not necessarily bounded into any specific interval of time. This type of content typically represents the "on-demand" component of a service. If this attribute is present and “true” it indicates that the referenced content item SHALL NOT be accessed automatically through this ‘Schedule’ fragment by the terminal when the associated service is selected and this ‘Schedule’ fragment is valid. This attribute in the 'Schedule' fragment MAY be set to ‘true’ if and only if the 'Access' fragments associated to the 'Schedule' fragment do not have the 'BroadcastServiceDelivery' element instantiated. In section 5.8.4.1.it is described how a ‘Content’ fragment can ‘inherit’ the ‘Access’ fragments that are associated with the ‘Service’ fragment. However, in the case the ‘onDemand’ attribute of a ‘Schedule’ fragment associated with a content item is ‘true’ the terminal SHALL assume that the ‘Access’ fragment associated with the ‘Service’ fragment is NOT applicable for a certain ‘Content’ fragment. In the case a ‘Content’ fragment has multiple ‘Schedule’ fragments associated with it, but only one or a subset of those ‘Schedule’ fragment have the ‘onDemand’ attribute set to ‘true’ the terminal SHALL ignore all ‘Schedule’ fragment that have the ‘onDemand’ attribute set to ‘false’ (either declared or by absence of it).

With regard to the combination of ‘defaultSchedule’ and ‘onDemand’ the following applies:

- for ‘Content’ fragments that have one or more ‘Schedule’ fragments associated with it that have the ‘onDemand’ attribute set to ‘true’, the ‘Schedule’ fragment with the ‘defaultSchedule’ set to ‘true’ determines the favourable access method of that content item. The terminal MAY use this information to facilitate the end user selection between the options of the various applicable access methods.
- for Content’ items’ that are only associated with ‘Schedule’ fragments that all have the ‘onDemand’ attribute set to false (either declared or by absence of it) the terminal SHALL interpret the access method represented by the ‘Schedule’ fragment to be applicable without possible end user intervention. In this case the ‘Schedule’ fragment associated with the ‘Content’ fragment that has the ‘defaultSchedule’ attribute set to true takes precedence. If none of the ‘Schedule’ fragment referring to the ‘Content’ fragment have the ‘defaultSchedule’ attribute set to ‘true’ the terminal SHALL use the ‘Access’ fragment that is associated with the ‘Service’ fragment as the default (see section 5.8.4.1).

5.8.6 Terminal interpretation for content consumption

When the terminal receives the complete Service Guide, the terminal will find it has a collection of ‘Service’ fragments and for each ‘Service’ fragment there is a set of ‘Content’ fragments, ‘Schedule’ fragments and ‘Access’ fragments associated with it. It is important to note that the elements and attributes provided by the ‘Service’ and ‘Content’ fragments provide information how particular content can be offered to the end user for consumption. The ‘Schedule’ and ‘Access’ fragments on the other hand provide the true availability and access methods of content. From a terminal point of view, whenever a service is selected by the end user, the terminal will find itself with a set of ‘Schedule’ and/or ‘Access’ fragments. This set of ‘Schedule’ and/or ‘Access’ fragments can be considered to have a time order where at a certain point in time multiple fragments can be valid. This is regardless of whether these ‘Schedule’ fragments do or do not have a specific reference to a ‘Content’ fragment associated with the corresponding service. Therefore whenever a service is selected the terminal has to access the content through the ‘Access’ fragment that is applicable at that particular point in time. Recall section 5.8.1.1, Choices of access.

At any point in time the terminal might find there are more ‘Access’ fragments valid for the service because of one or more of the following use cases:

- Both unicast and broadcast accesses are available for the service,
- multiple text and/or audio languages are available for the service or

- multiple text and/or audio languages are available for a specific piece of content at that point in time.

The consumption of a single piece of content that is part of a service can made available for the end user in the following ways:

- automatically without user intervention upon entering the service by terminal selection of the access method that is valid at the point in time, based on the content items that are part of the regular programming, as indicated by Schedule fragments that have the 'onDemand' attribute set to 'false' (either declared or by absence of it)
- upon user selection.

In the first case terminal SHALL only be required to make a selection from 'Access' fragments that are associated:

- with the 'Service' fragment as described in section 5.8.2.
- with the 'Service' fragment through a Schedule fragment as described in section 5.8.3.
- with 'Schedule' fragments related to 'Content' fragments that that are not part of the "on demand" content, which are defined by having the 'onDemand' attribute set to 'false' (either declared by absence of it).

to determine which 'Access' fragment is valid and takes precedence and MAY be used for automatic content consumption.

In the latter case, access based on user selection is applicable for:

- 'Content' fragments that have the 'Schedule' fragment related to it with the 'onDemand' attribute set to 'true'
- 'Content' fragments that have 'Schedule' fragments related to it that have the both the 'onDemand' and the 'defaultSchedule' attributes set to false (either declared or by absence of it).

Whenever there are multiple text or audio languages available the terminal can give the end user the option to make a selection. The terminal can for example highlight or place blinking icons with the content in the Service Guide that has at that point in time an applicable access associated with it. Based on the specific terminal language settings or based on the 'Schedule' fragment that is declared to have the 'defaultSchedule' attribute set to 'true' the terminal can make a decision which 'Access' fragment provides the preferable access parameters for the service.

It is important to note that the 'StartTime' and 'EndTime' elements in the 'Content' fragment do not declare any validity of the content whatsoever, these elements are purely intended to present to the end user enabling the terminal to tell the end user which piece of content is currently being played. In a typical case the network could declare a set of 'Content' fragments associated with the same 'Service' fragment, with the 'StartTime' element and 'weight' attribute set to represent a chronological order of pieces of content belonging to a service. The 'StartTime' of a 'Content' fragment could then implicitly serve as the 'EndTime' of the 'Content' fragment with the next lower 'weight' attribute. For the cases above it is very well possible that a certain 'Access' fragment that is associated with a 'Content' fragment, is applicable during an interval that mismatches with the time window, as defined by the 'StartTime' and 'EndTime' elements (either explicit or implicit declared), of that 'Content' fragment. For instance it could have a partial overlap of a time window of other 'Content' fragments belonging to the same service.

Despite the Service Guide having coherent structure there can be three cases in which the terminal will encounter unexpected events:

- In the event that by accident the Service Guide does have overlapping 'Schedule' fragments with the'onDemand' attribute set to 'false' (either declared by absence of it) and that are associated to different 'Content' fragments the terminal might not be able to determine the 'Access' fragment that is valid and should be selected, i.e. multiple 'Access' fragments are valid and the terminal can not uniquely decide which one takes precedence over the other based on the 'defaultSchedule' attribute. In such a case the terminal MAY assume that of these overlapping 'Schedule' fragments the 'Schedule' fragment which has the 'PresentationWindow' element with the earliest 'startTime' attribute to take precedence and accordingly the 'Access' fragment associated with this 'Schedule' fragment.
- In the event that a certain 'Access' fragment is selected but there is no IP stream available on the declared IP address in the 'SessionDescription' fragment. This can e.g. be the case when the live broadcast session schedule has

changed, but the Service Guide could not be updated, or just because the access is currently unavailable e.g. due to radio reception conditions. In such a case the terminal MAY select another applicable 'Access' fragment. In case no such 'Access' fragment is available the terminal MAY present an error to the end user. In any case the terminal SHOULD be able to handle such a case and not turn to an error-state.

- In the event that a 'Content' fragment has multiple text and audio languages declared, where all audio and text languages are associated with 'Schedule' fragments that directly reference that 'Content' fragment, but none of these ''Schedule' fragments have the 'defaultSchedule' attribute set to 'true' and none of these 'Schedule' fragments have the 'onDemand' attribute set to 'true' and 'none' of the languages match the specific terminal language settings. In this case the terminal SHALL use the 'Access' fragment that is associated with the 'Service' fragment (which is referenced by that 'Content' fragment, see section 5.8.4.1).
- In the event a 'Content' fragment has multiple 'Schedule' fragments associated with it, but only one or a subset of those 'Schedule' fragment have the 'onDemand' attribute set to 'true'. In this case the terminal SHALL ignore all 'Schedule' fragment that have set the 'onDemand' attribute to 'false' (either declared or by absence of it).

5.8.7 Distribution Window and Presentation Window for Cachecast Services

In addition to streaming services, for which the rendering time is the same as the delivery time, the SG also supports file delivery services for which the network can distribute the content prior to rendering. For file distribution services, the terminal downloads the content beforehand and stores it for later user consumption. Two types of file delivery services are possible: those which pertain to scheduled rendering, such as Cachecast, and others which are not associated with scheduled rendering, such as music or video downloads. The temporal semantics of Cachecast services are specified by the Distribution Window and Presentation Window elements in the Schedule fragment. Note in particular that for Cachecast, neither window is necessarily fixed in time or size:

- The terminal can begin the download as early as 'startTime' of Distribution Window; the 'startTime' does not represent the default start time of the download.
- The terminal can begin the download as late as 'endTime' of Distribution Window and should still be able to acquire the entire file within the 'duration' of the Distribution Window; the 'endTime' does not represent the default end time of the download.
- The user can begin rendering as early as the 'startTime' of the Presentation Window; the 'startTime' does not represent the default start time of rendering;
- The user can begin rendering as late as the 'endTime' of the Presentation Window, and still be able to expect the rendering to run to completion; the 'endTime' does not represent the default end time of rendering. The presentation duration is given by the 'duration' of Presentation Window.

6. Discovery of Service Guide

6.1 Service Guide Transmitted over Broadcast Channel

The service guide discovery mechanisms that are specified in this section relate to the discovery of a Service Guide that is distributed over Broadcast Channel (i.e. they do not relate to Service Guides that are not using the Broadcast Channel). In this case, the Terminal will need to know how to find and access the broadcast IP flows that carry the broadcast Service Guide.

6.1.1 Service Guide Discovery over Broadcast Channel

When the Service Guide is delivered using the broadcast channel the Service Guide Announcement Channel is thought as the starting point of the retrieval. Recall that the Service Guide Announcement Channel provides all the information the terminals need for retrieving the Service Guide. Therefore to discover the Service Guide the terminals basically need to locate the file delivery session carrying the Service Guide Announcement Channel. The access parameters of the FLUTE session representing the Service Guide Announcement Channel are called the entry point to a Service Guide on a Broadcast Channel.

In one broadcast area there MAY exist multiple Service Guides and any number of these MAY be delivered simultaneously using the broadcast channel. In such a case, in principle, it is the responsibility of the underlying BDS to provide the signalling of the entry points of the Service Guides to the terminals. However, if such a signalling is not available or being used, the following parameters SHALL be used as the entry point:

- (OPTIONALLY) IP Source Address
- Fixed Destination Multicast IPAddress: 224.0.23.165 for IPv4 or FF0X:0:0:0:0:0:0:132 for IPv6
- Fixed Destination Port: 4090

The terminal SHALL assume that

- there is at most one FLUTE session per entry point.
- the value of the Transport Session Identifier (TSI) MAY be any valid value and the number of ALC/LCT channels in the FLUTE session for Service Guide announcement is fixed to 1.

If the underlying BDS supports specific signalling of the Service Guide entry points the terminal SHALL expect the BDS also to provide the specific signalling. The detailed guidelines for such signalling in specific Broadcast Distribution Systems are given in the BDS Adaptation Specifications (See [BCAST10-DVBH-IPDC-Adaptation], [BCAST10-MBMS-Adaptation] and [BCAST10-BCMCS-Adaptation]).

The terminal SHALL support the initial Service Guide discovery over Broadcast Channel.

6.1.2 Service Guide Discovery over Interaction Channel

The entry point to a Service Guide on an Interaction Channel SHALL be defined as the URL to a file containing Session Description or URL to a resource resolving to a Session Description which describes the file distribution session carrying Service Guide announcement information and possibly Service Guide. This file distribution session originates from Service Guide Generation Function and Service Guide Distribution Function (over interface SG-5). The entry point to a Service Guide on an Interaction Channel MAY be either fixed, or provisioned to the terminal (e.g. through BDS specific signalling), or provided out-of-band (e.g. through a public or private web site).

Within a single BDS, there MAY be different Service Guide generated for different service coverage areas, requiring a different entry point for each particular service coverage area. It is not in scope of this specification to define how the device learns about the applicable URL.

The terminal with interaction channel SHALL support the initial Service Guide discovery over Interaction Channel.

6.2 Service Guide Transmitted over Interaction Channel

The service guide discovery mechanisms that are specified in this section relate to the discovery of a Service Guide that is to be distributed over Interaction Channel. The Terminal needs to get some discovery information, and sends the request to acquire Service Guide.

The entry point to Service Guide acquisition over Interaction Channel SHALL be a URL which indicates the location of Service Guide. Example of such URL is (http://provider.com/serviceguide) .This is the address that the SG-C in the Terminal accesses in order to get Service Guide data over Interaction Channel as specified in section 5.4.3. There are several possible ways terminal can get the entry point information. The Terminal SHALL support the following two means: the entry point information is provided using the 'AlternativeAccessURL' element of SGDD; and; the entry point information is provisioned to the Terminal via Terminal Provisioning function. For the latter case the terminal SHALL support OMA BCAST Management Object parameter '/<X>/SGServerAddress/'as specified in [BCAST10-Services]. Further the entry point information MAY be fixed in the Terminal or provided out-of-band by the means not specified in this specification (such as WAP PUSH, SMS, MMS, Web page, user input, etc).

7. Multi-language Support

This section defines how the language information conveyed in the Service Guide can be used both to specialize the presentation of the Service Guide itself for individual languages and to provide language specific options for consuming the services declared by the Service Guide. Whereas the former is in principle about textual descriptions provided in individual languages, the latter also considers possible language dependent audio tracks and sub-titles. The section is organized to address the specialization of the Service Guide regarding how to declare the language specific options of consuming the services. It should be noted that the technical details of enabling multi-language support are provided in conjunction with the corresponding attributes and elements of the Service Guide and that the following sections merely provide logically meaningful binding of these individual details.

7.1 Declaring service guide elements in different languages in the Service Guide

The Service Guide may provide both technical information not visible to the end user and purely descriptive information meaningful only to the end user. Whereas the former part is independent of such things as the nationalities of the end users, the latter can be explicitly specialized for end users with different nationalities. The data model of the Service Guide enables the network to provide textual descriptions in the Service Guide multiple times, one description per language. This is simply achieved by allowing an individual element representing a textual description to be instantiated multiple times and specializing each of the individual instances of these elements using the 'xml:lang' attribute defined by XML standard itself.

In case an individual element of the Service Guide represents textual description intended only for the UI purposes the following applies:

- In case the element is instantiated more than once in an individual SG fragment, the network MUST include the 'xml:lang' attribute to each of the instances of the element in the SG fragment where
- each of the instantiated 'xml:lang' attributes MUST have a different value.
- On the other hand if the 'xml:lang' attribute is omitted in the element the terminal SHALL interpret the element to be applicable independent of any language settings.

Table 5 below summarizes the elements that can be specialized for different languages.

	Service	Content	Purchase Item	Purchase Channel	Purchase Data	Preview Data	Interactivity Data	SGDD
Name	X	X	X	X	X			
Description	X	X	X	X	X			
Extension.Description	X	X	X	X	X		X	
AudioLanguage	X	X						
TextLanguage	X	X						
Genre	X	X						
BroadcastArea.TargetArea. name_area	X	X						
PromotionInfo.Title					X			
PromotionInfo.Description					X			
PromotionInfo. PromotionExtension. Description					X			
PromotionInfo.Title					X			
Video.AlternativeText						X		
Audio.AlternativeText						X		
Picture.AlternativeText						X		
Text						X		
InteractivityType							X	

BSMList.BSMSelector.Name								X
DescriptorEntry.GroupingCriteria. GenreGroupingCriteria								X

Table 5: Elements that can be specialized for different languages

7.2 Declaring components of the broadcast service in different languages in the Service Guide

The services, declared by 'Service' fragments, as well as individual parts of the services, declared by 'Schedule' and 'Content' fragments, are possible to be provided in multiple ways to be consumed. Language is one possible criterion for end users to differentiate and consume various services. In order for the end users to understand the meaning of the different options, the Service Guide needs to be able to represent the criterion as well as the individual choices for the end user. For this purpose the Service Guide provides specific elements and attributes that can be used to signal the semantics of the different choices, in this case the specific languages.

7.2.1 Service and content specialization with audio tracks and sub-titles

The network can declare both services as well as individual pieces of services, represented by a 'Service' and 'Content' fragments respectively, to be available in different languages

- with multiple audio streams and/or
- with multiple sub-title streams.

In the Service Guide the available language is declared by the 'AudioLanguage' and/or 'TextLanguage' elements of the 'Content' and 'Service' fragments respectively. The association to the specific details of consumption is in turn achieved through 'audioLanguageIDRef' and 'textLanguageIDRef' attributes of the 'ContentReference' and 'ServiceReference' elements in the 'Schedule' fragment. Similarly as in the case of language specialization of the SG, it is necessary to have common understanding between the network and the terminals on the semantics of the multiple choices of consumption. For enabling the correct interpretation of the language choices

- the terminal SHALL interpret any reference from a 'Schedule' fragment to a 'Content' or 'Service' fragment without either of the attributes 'audioLanguageIDRef' and 'textLanguageIDRef' to be applicable independent of any language settings and
- the terminal SHALL interpret an 'Access' fragment referring to a 'Schedule' fragment having either or both of the attributes 'audioLanguageIDRef' and 'textLanguageIDRef' set as providing the consumption details for the specific language choice represented by the 'Schedule' fragment.

Furthermore if a 'Schedule' fragment contains the 'defaultSchedule' attribute the terminal

- SHALL interpret the 'Schedule' fragment as giving default consumption details regardless of any individual language settings when both the attributes 'audioLanguageIDRef' and 'textLanguageIDRef' are omitted, but
- if either of the 'audioLanguageIDRef' or 'textLanguageIDRef' attributes are present, the terminal SHALL interpret the 'Schedule' fragment as giving the consumption details only for those languages that are fixed by these attributes.

Recalling that a 'Content' fragment can be associated with a 'Schedule' fragment either

- directly by the 'Schedule' fragment referring the 'Content' fragment or
- indirectly the 'Schedule' fragment referring to the 'Service' fragment the 'Content' fragment is associated with

Therefore the terminal SHALL first consider the 'Schedule' fragments associated directly with the 'Content' fragment and if none of them is applicable only then consider any of the indirectly associated 'Schedule' fragments.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version –or- No previous version within OMA
OMA-xxyyz-V1_0-20021001-A	01 Oct 2002	Initial document to address the basic starting point Ref TP Doc# OMA-TP-2002-1234-xxyyzForApproval
OMA-xxyyz-V1_1-20030405-A	05 Apr 2003	description of changed Ref TP Doc# OMA-TP-2003-0321-xxyyzV1_1forApproval

A.2 Draft/Candidate Version 1.0 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-TS_BCAST_ServiceGuide-V1_0	xx Jan 2005	n/a	Initial Draft
	17 Jan 2005	5, 6	Agreed Draft
	26 Jan 2005	5.4	CR agreed during BCAST interim meeting in Seoul: OMA-BCAST-2005-0009R01-CR-TS-ServiceGuide-Delivery
	04 Feb 2005	5.2.1 5.4.3 6.1	CRs agreed during BCAST meeting in Frankfurt: OMA-BCAST-2005-0008R01-CR-TS-ServiceGuide-Data-Model OMA-BCAST-2005-0035R01-LATE-New-Service-Guide-Delivery-Unit OMA-BCAST-2005-0040R01-LATE-CR-TS-SG-Bootstrapping
	29 Mar 2005	5.2.1 5.4.3	CRs agreed during BCAST meeting in Chicago: OMA-BCAST-2005-0109-LATE-CR-TS-ServiceGuide-DataMode OMA-BCAST-2005-0084-CR-GPP2_Metadata_Envelope
	06 May 2005	5.2.2	CRs agreed during BCAST meeting in Singapore: OMA-BCAST-2005-0143R01-CR-BCAST-TS-SG-Preview_Data_in_Service_Guide
	12 May 2005	5.2.2	CRs agreed during BCAST meeting in Singapore: OMA-BCAST-2005-0129R02-CR-Service-Guide-data-model-Access-Fragments OMA-BCAST-2005-0130R03-CR-Service-Guide-data-model-Core-Fragments
	03 June 2005	5.2.2 5.4	CRs agreed during BCAST CC on May 18 2005: OMA-BCAST-2005-0148R02-CR-BCAST-TS-SG-Location_information_in_Service_Guide CRs agreed during BCAST meeting in Singapore: OMA-BCAST-2005-0131R04-CR-Service-Guide-Datamodel-Provisioning-Fragments OMA-BCAST-2005-0180R03-CR-TS-Service-Guide-Delivery-Descriptor OMA-BCAST-2005-0213-CR-BCAST-TS-SG-Bandwidth_Requirement OMA-BCAST-2005-0214-CR-BCAST-TS-SG-Alternative_Access_URL OMA-BCAST-2005-0215-CR-BCAST-TS-SG-Content_Timing

Document Identifier	Date	Sections	Description
	22 June 2005	7 6.2 5.4.5 5.5 5.2.2 5.1.1 5.4.2 Appendix C 5.2.1	CRs agreed during BCAST meeting in San Diego: OMA-BCAST-2005-0239-MIME-media-type-of-Service-Guide OMA-BCAST-2005-0240R03-LATE-Service-Discovery-Mechanisams OMA-BCAST-2005-0241R01-LATE-Service-Guide-Update-and-Management OMA-BCAST-2005-0242R02-LATE-CR-Service-Guide-Service-Preview OMA-BCAST-2005-0244R01-add-column-for-datatype-in-BCAST-tables OMA-BCAST-2005-0252R01-CR-BCAST-TS-SG-Promotion_Info OMA-BCAST-2005-0253R01-CR-BCAST-TS-SG-Target_User_Profile OMA-BCAST-2005-0254-CR-BCAST-TS-SG-terminology OMA-BCAST-2005-0257R01-Updated-SGDD-definition OMA-BCAST-2005-0258R01-Streamlining-SG-Core-fragments OMA-BCAST-2005-0259R01-Attributes-common-to-all-Service-Guide-fragments OMA-BCAST-2005-0265R02-Proposed-initial-Service-Guide-XML-Schema OMA-BCAST-2005-0270R01-Streamlining-SG-Provisioning-fragments OMA-BCAST-2005-0271R01-Clean-up-of-Service-Guide-Data-Model-Figure OMA-BCAST-2005-0274-LATE-CR-clarification-on-cardinalities OMA-BCAST-2005-0276R01-LATE-Service-Guide-access-fragment- OMA-BCAST-2005-0283-LATE-Service-Guide-Bugfix OMA-BCAST-2005-0284-SG-Diagram
	11 July 2005	5.2.2	CRs agreed during BCAST Conference Call on July 6 2005 OMA-BCAST-2005-0296R01-CR-SG-Datatypes OMA-BCAST-2005-0301R01-CR-SG_Change_Proposal_for_Distribution_and_Presentation_Window
	05 Aug 2005	2.1 5.2.2.2 5.2.2.4 5.2.2.9 5.4.1.1 5.4.2	CRs agreed during BCAST Conference Call on Aug 3 2005 OMA-BCAST-2005-0261R02-Proposed-change-introducing-ALC-as-Service-Guide-delivery-protocol OMA-BCAST-2005-0277R01-LATE-Add-MediaInformation-Element-to-ScheduleItem OMA-BCAST-2005-0292R05-Enhancements-to-SGDD OMA-BCAST-2005-0297R02-CR-SG-AccessType-and-AssociateDeliveryProc OMA-BCAST-2005-0309R02-G_Change_Proposal_for_autostart_time_info OMA-BCAST-2005-0317R01-CR-SG-PreviewData-Picture
	18 Aug 2005	5.2.2.3 Appendix C	CRs agreed during BCAST Conference Call on Aug 17 2005 OMA-BCAST-2005-0282R03-XML-Schema-for-SGDD OMA-BCAST-2005-0304R02-CR-SG-File-Signalling

Document Identifier	Date	Sections	Description
	02 Sep 2005	2.1 3.2 5.2.1 5.2.2 5.4.1.1 5.4.2 5.4.5 6.1 Appendix E	CRs agreed during OMA BCAST Montreal meeting: OMA-BCAST-2005-0298R02-CR-SG-ServiceType OMA-BCAST-2005-0312R02-CR-SGDD_Clarification OMA-BCAST-2005-0315R01-LATE-ALC_for_ESG_Delivery OMA-BCAST-2005-0332R03-Change-the-DW-and-PW OMA-BCAST-2005-0334-CR-SG-Content-Type OMA-BCAST-2005-0335-CR-SG-PreviewData-XML-Schema OMA-BCAST-2005-0352R01-SessionDescription-for-file-delivery-session-+-corrections OMA-BCAST-2005-0353-XML-Schema-for-provisioning-fragments OMA-BCAST-2005-0354-language-and-xml-lang-in-SG OMA-BCAST-2005-035501-Clarification-related-to-entry-point-of-SG-discovery OMA-BCAST-2005-0356-Service-Guide-Delivery-bug-fix-and-clarification OMA-BCAST-2005-0357R01-Clarification-of-ExtensionURL OMA-BCAST-2005-0358-Clarification-on-the-use-of-GroupingCriteria OMA-BCAST-2005-0359R01-Proposal-to-use-32-bit-integer-for-fragment-id-and-version OMA-BCAST-2005-0361R01-CR-SG-Data-Maintenance-and-Update OMA-BCAST-2005-0368R02-CR-SG_Identification_of_DW_and_PW_in_the_Shedule_Item OMA-BCAST-2005-0370R01-CR-SGDD-Identification_of_Service_Provider OMA-BCAST-2005-0373R01-CR-SG-bug-fix-AccessURI-and-SessionDescription CRs agreed during the Conference Call on Aug 31 2005: OMA-BCAST-2005-0389R01-Fixing-Cardinalities-Service-Guide-Figure
	10 Sep 2005	5.2.2.4	CR agreed during the Conference Call on Aug 31 2005: OMA-BCAST-2005-0383R03-CR-Access_Relation_to_Schedule
	30 Sep 2005	5.2.1 5.2.2 5.3.1 5.4.2 5.4.4 Appendix C	CR agreed during the Conference Call on Sep 14 2005: OMA-BCAST-2005-0364R02-CR-TS-Service-Guide-Clarification-on-Access-Fragment OMA-BCAST-2005-0386R03-Service-Guide-fragments-descriptions CRs agreed during OMA BCAST Tokyo interim meeting: OMA-BCAST-2005-0392R01-CR-Service-Guide-TS-FragmentID-URI-Mapping OMA-BCAST-2005-0402R01-Specification-of-the-Genre OMA-BCAST-2005-0414-Bug-fix-for-the-content-fragment OMA-BCAST-2005-0421R04-CR-Ordering_Content_Appearance_on_Display OMA-BCAST-2005-0437R02-Clarification-on-the-signalling-of-FEC OMA-BCAST-2005-0440R01-CR-SG-category-and-cardinality-of-Description-elements OMA-BCAST-2005-0442R01-CR-SG-changes-on-PurchaseData-fragment OMA-BCAST-2005-0444R01-CR-SG-changes-on-RightsIssuerId-in-PurchaseChannel OMA-BCAST-2005-0445-CR-SG-update-of-XML-schema OMA-BCAST-2005-0447-CR-SG-SGDD-Delivery OMA-BCAST-2005-0450R01-CR-SG-Backend-Interfaces-related-SCR-tables OMA-BCAST-2005-0453-CR-MBMS-USD-in-SG

Document Identifier	Date	Sections	Description
	26 Oct 2005	2.2 5.2.1 5.4 7 8 Appendix D	CR agreed during the Conference Call on Oct 5 2005: OMA-BCAST-2005-0456-CR-SG-link-between-PreviewData-and-Schedule OMA-BCAST-2005-0458R01-Adding-Arrowheads-to-SG-Data-Model-Figure CRs agreed during OMA Sydney meeting: OMA-BCAST-2005-0406R04-Interaction-fragment-for-the-service-guide OMA-BCAST-2005-0407R01-CR-DW-and-PW-fragment-for-the-service-guide OMA-BCAST-2005-0412R02-Modification-of-SGDD-for-Notification-Function OMA-BCAST-2005-0413R02-Notification-Section-for-TS-Service-Guide OMA-BCAST-2005-0423R03-CR-SG_Changes_For_Open-and-Close-State OMA-BCAST-2005-0426R06-CR-Inclusion_of_PurchaseItems_by_Reference OMA-BCAST-2005-0431R01-Key-Management-Signalling OMA-BCAST-2005-0451R02-CR-SG-Change-mgmt-over-broadcast OMA-BCAST-2005-0452R01-CR-SG-Completeness OMA-BCAST-2005-0492R03-CR-SG-Cell-ID-based-broadcast OMA-BCAST-2005-0498R01-CR-SG-MIME-types-for-SG OMA-BCAST-2005-0500R01-CR-SG-Bring-back-missing-Service-URI OMA-BCAST-2005-0511R01-Add-note-to-clarify-cardinalities-in-SG-figure OMA-BCAST-2005-0519R02-CR-SG-Terminal-Capability OMA-BCAST-2005-0520-CR-SG_Changes_for_MediaInformation OMA-BCAST-2005-0521-CR-SG-Scenario-for-codec-based-selective-viewing OMA-BCAST-2005-0522R02-CR-SG-Application-Scenario-for-bandwidth-based-scalable-viewing OMA-BCAST-2005-0525R01-CR-SGDD-for-interactive-SG-retrieval OMA-BCAST-2005-0531-CR-Program-Rating OMA-BCAST-2005-0552R01-CR-SG-Notification-service-type

Document Identifier	Date	Sections	Description
	29 Nov 2005	3.3 5.2.1 5.2.2 5.3 5.5.2.2 5.6 8.1 8.2.2 8.3 8.4 Appendix F	CR agreed during the Conference Call on 2 Nov 2005: 2005-0422R04-CR-SG_Blackout_Service_Reception_Areas CRs agreed during OMA BCAST Madrid interim meeting: OMA-BCAST-2005-0643R01-CR-Prelisten-Behaviour OMA-BCAST-2005-0637R01-CR-SG-Service-Description OMA-BCAST-2005-0635R01-CR-Clarifications-on-purchase-channel-data OMA-BCAST-2005-0609R01-CR-SG-Associated-Delivery-Procedures OMA-BCAST-2005-0597-TS-SG-Interface-SG1 OMA-BCAST-2005-0596R01-TS-SG-Interface-NT4 OMA-BCAST-2005-0595R01-TS-SG-Interface-NT3 OMA-BCAST-2005-0593R02-TS-SG-Interface-NT1 OMA-BCAST-2005-0592R01-TS-SG-refining_section_8_2_2 OMA-BCAST-2005-0587R03-SG-target-user-profile OMA-BCAST-2005-0586-SG-datatype-in-XMLSchema OMA-BCAST-2005-0585R02-SG-bug-fix-of-InteractivityData-fragment OMA-BCAST-2005-0584R01-clarification-on-arrow-direction-in-SG-Data-Model-Figure OMA-BCAST-2005-0581-CR_Clarification_to_Presentation_Window OMA-BCAST-2005-0580R02-CR_Adding_Duration_to_Distribution_Window OMA-BCAST-2005-0575R01-CR-SG-Referred-ServiceGuides OMA-BCAST-2005-0548-CR-SG-Terminal-Provisioning-service-type OMA-BCAST-2005-0523R03-CR-Availability_of_PurchaseItems OMA-BCAST-2005-0501R01-Categorization-of-SG-elements-and-attributes OMA-BCAST-2005-0471R01-CR-SG-cardinality-between-content-and-service
	06 Jan 2006	2.1 3.3 5.2.1 5.2.2 5.5.1.2 5.5.2.2 5.5.5 5.6.2 5.6.3 8 Appendix D.	CRs agreed during OMA BCAST Athens meeting: OMA-BCAST-2005-0755R01-Include-user-start-and-end-time-in-the-Program-guide-info-for-the-ESG OMA-BCAST-2005-0739R02-CR-location-based-broadcast OMA-BCAST-2005-0709-CR-SG-Notification-Cleanup OMA-BCAST-2005-0708R01-CR-remove-type-in-SGDD OMA-BCAST-2005-0686R01-CR-Additional_Radio_Related_Service_Types OMA-BCAST-2005-0678-CR-SG-framgents-update-and-management OMA-BCAST-2005-0677R04-CR-missing-categorization-in-Schedule-fragment OMA-BCAST-2005-0676R01-CR-SG-TS-Abbreviation OMA-BCAST-2005-0669-GlobalContentID-SG OMA-BCAST-2005-0666R01-Zapping_support OMA-BCAST-2005-0614R04-Program-guide-info-for-the-ESG OMA-BCAST-2005-0599R01-TS-SG-Interface-SG4 OMA-BCAST-2005-0598R01-TS-SG-Interface-SG2 OMA-BCAST-2005-0555R03-CR-BDS-indicators-in-SG OMA-BCAST-2005-0526R02-CR-HTTP-request-format-for-interactive-SG-retrieval

Document Identifier	Date	Sections	Description
	15 Feb 2006	2.1 5.2.1 5.2.2 5.4.1.1.3 5.5.1.1 5.5.2.2 5.5.4 5.6.1.2.1 5.7 8. 8.2 8.3 Appendix B Appendix F	CRs agreed during OMA BCAST Review and Approval Process 05-18 Jan 2006: OMA-BCAST-2005-767R01-Enhancement ESG Delivery OMA-BCAST-2005-0754R01-Selector-Type OMA-BCAST-2005-0705R01-CR-InteractivityMedia-over-separate-access CRs agreed during OMA BCAST Paris meeting: OMA-BCAST-2006-0131-Compact-No-Code-for-SG-delivery-over-broadcast-channel OMA-BCAST-2006-0127-Clean_up_the_sentences_related_with_ServiceClass OMA-BCAST-2006-0124R02-TS-SG-Auxiliary-Data-Trigger OMA-BCAST-2006-0111-CR-clarification-on-backend-interfaces OMA-BCAST-2006-0106-CR-SG-Service-Access-Relation OMA-BCAST-2006-0094R01-CR-SG_Bug_Fixes OMA-BCAST-2006-0076R01-CR-SG-PreviewData-Uses OMA-BCAST-2006-0071R01-CR-Interface-SG-1 OMA-BCAST-2006-0070R01-CR-SG-and-Notification-SCR-tables OMA-BCAST-2006-0063R01-CR-SG-RepeatType OMA-BCAST-2006-0062-Data-types-of-zip_code-and-PhoneNumber OMA-BCAST-2006-0061R01-Proposed-Changes-to-SGDD OMA-BCAST-2006-0057R03-CR-Parameters-for-Session-Description OMA-BCAST-2006-0042-Using-unsignedByte-for-Type OMA-BCAST-2006-0041R01-Deletion-of-ServiceProtection-in-Service-Fragment OMA-BCAST-2006-0040R01-Cleanup-in-PurchaseItem-and-PurchaseChannel OMA-BCAST-2006-0038-Adding_a_type_of_ServiceClass_in_SG OMA-BCAST-2006-0035-Clean-up-of-Content-Fragment OMA-BCAST-2006-0034-Bug-fix-in-Schedule-Fragment OMA-BCAST-2006-0031-CR-SG-Fragment-Encoding-Field-SGDU OMA-BCAST-2006-0007R01-CR-Content-type-clarification OMA-BCAST-2005-0616R02-CR-SG-Interaction-Service-Type OMA-BCAST-2005-0513R04-Service-Guide-Context

Document Identifier	Date	Sections	Description
	22 Mar 2006	1, 2. 3. 4. 5. 7. 8. Appendix C. Appendix F. Appendix G. Appendix H.	**CR agreed during the Conference Call on 25 Jan 2006:** OMA-BCAST-2006-0017-CR-SG-ApplicationSpec-to-ServiceClass-Bug-Fix **CRs agreed during OMA BCAST Seoul interim meeting:** OMA-BCAST-2006-0254-CR-SG-Scope-and-Introduction OMA-BCAST-2006-0250-CR-Bugfix-on-Codec OMA-BCAST-2006-0249R01-CR-initial-buffering-delay-in-SDP OMA-BCAST-2006-0246R01-Notification-Subscription OMA-BCAST-2006-0245R01-SG-BackendInterface OMA-BCAST-2006-0244R01-NT-BackendInterface OMA-BCAST-2006-0238R02-CR-refine-SplitTOI-section OMA-BCAST-2006-0237R03-CR-SG-Content-Encoding-Signalling-for-SGDU OMA-BCAST-2006-0235R01-Clarification-text-for-SG-Delivery-and-Request OMA-BCAST-2006-0233-CR-SG-Cleanup OMA-BCAST-2006-0232R01-Status-Codes-TS-Service-Guide OMA-BCAST-2006-0231R01-SG-MIMETypes-Namespaces OMA-BCAST-2006-0227-Add-text-language-information-in-SG OMA-BCAST-2006-0225R01-CR-harmonized-BCAST-crossreferences OMA-BCAST-2006-0216-CR-removal-of-bearer OMA-BCAST-2006-0215-CR-offset-in-SGDU OMA-BCAST-2006-0200R03-CR-Revised_SG_Update_Detection_Flowchart OMA-BCAST-2006-0199R01-Notificaiton-ServiceClass OMA-BCAST-2006-0197R01-CR-time-based-interactive-ESG-request OMA-BCAST-2006-0196R02-CR-Some-Fixes-in-Access-Fragment OMA-BCAST-2006-0193R03-CR-ESG-Unicast-support OMA-BCAST-2006-0192R04-CR-Adding_SGDU_Version_to_SGDD OMA-BCAST-2006-0184-CR-Simplify-Messages-over-Notification-Interfaces OMA-BCAST-2006-0182R01-Clarification-on-Aux-trigger-support OMA-BCAST-2006-0180-CR-SG-Global-Purchase-Item-Id OMA-BCAST-2006-0177R03-CR-Structure_of_FilteringData_in_AuxDataTrigger OMA-BCAST-2006-0175R01-CR-Reinstate_Mixed_RT-NRT_ServiceTypes OMA-BCAST-2006-0173R01-CR-Notification-Bug-Fix OMA-BCAST-2006-0172R02-CR-PreviewData-usage OMA-BCAST-2006-0171R01-CR-refine-SplitTOI-section OMA-BCAST-2006-0170-CR-errors-of-multiple-attritbutes OMA-BCAST-2006-0161-CR-SG-Fragments-Schema-Update OMA-BCAST-2006-0160-CR-SGDD-Schema-Update OMA-BCAST-2006-0159-Terminal-Binding-use-in-ESG OMA-BCAST-2006-0154-Consistent--WindowID-types-in-SG OMA-BCAST-2006-0141R01-SG-Delivery-and-Request-Btw-BSM-and-BSDA OMA-BCAST-2006-0043R02-Clean-up-of-PriceInfo OMA-BCAST-2005-0697R04--SG-sdp **CR agreed during the Conference Call on 15 March 2006:** OMA-BCAST-2006-0257R01-CR-Aux_Data_Distribution_Wrap-up
	24 Mar 2006	Appendix A	1. Changed the year in all copyright footer and copyright text from "2005"/"2004" to "2006". 2. Corrected the section indexes of 5.6.1, 5.6.2, 5.6.3 3. Corrected some editorial errors in Change History
	22 Dec 2006	all	Incoporated the resolutions of 1st round of BCAST1.0 consistency review comments

Document Identifier	Date	Sections	Description
	04 Jan 2007	all	**CRs agreed in Washington meeting but missed in OMA-TS_BCAST_ServiceGuide-V1_0_0-20061222-D:** OMA-BCAST-2006-0928R09-CR_Making_Roaming_Consistent_in_BCAST_1_0
	22 Mar 2007	all	Incoporated the resolutions of follow-up review comments of BCAST1.0 consistency review
	27 Mar 2007	all	Move Notification function related content to TS Services, and do corresponding changes. A few editorially clean-up.
	29 Mar 2007	All	Cleanup in preparation for Approval as Candidate
	04 April 2007	All	**Bug-fix CRs agreed over email:** OMA-BCAST-2007-0429R01-CR-SG-SessionDescription-Fix OMA-BCAST-2007-0430R01-CR_ValidTo_and_ValidFrom_in_SGDU OMA-BCAST-2007-0166R02-CR_Definition_of_RI_Service_service_type_in_the_Service_Guide In adtion, a few other editorials are made.
	05 April 2007	All	**CRs agreed in BCAST CC on 04 April 2007** OMA-BCAST-2007-0441-CR_SG_Fragments_Datamodel_Bugfix OMA-BCAST-2007-0449-CR_SG_Backend_Bugfix In adtion, a few other editorials are made.
	19 April 2007	All	**CRs agreed by email R&A from 10-12 April 2007:** OMA-BCAST-2007-0453-CR_TS_SG_update_section_5_5_1_1 OMA-BCAST-2007-0407R03-CR_TS_Distribution_Stream_Creation_Correction OMA-BCAST-2007-0464-CR_clarify_software_managment_services **CRs agreed by BCAST/DLDRM joint CC on 11 April 2007:** OMA-BCAST-2007-0418R02-CR_SG_protection_against_changes **CRs agreed in OMA Frankfurt meeting from 16-19 April 2007:** OMA-BCAST-2007-0462-CR_clerical_error OMA-BCAST-2007-0463-CR_spelling_error OMA-BCAST-2007-0443-CR_Blackout_declaration_in_SG_alternative_using_PreviewData OMA-BCAST-2007-0491R03-CR_TS_SG_SCR_Tables OMA-BCAST-2007-0509R01-CR_optional_attributes_defaults In adtion, a few other editorials are made.
	21 April 2007	All	**CRs agreed on Tursday of Frankfurt meeting:** OMA-BCAST-2007-0498R02-CR_Uniqueness_of_identifiers OMA-BCAST-2007-0512R01-CR_Distribution_FEC_SDP_Corr
	27 April 2007	5.1.2	**Bug-fix for cross-reference**
Candidate Version OMA-TS_BCAST_ServiceGuide-V1_0	29 May 2007	n/a	Status changed to Candidate by TP TP ref# OMA-TP-2007-0129R01-INP_BCAST_V1_0_ERP_for_Candidate_approval

Appendix B. Static Conformance Requirements (Normative)

The notation used in this appendix is specified in [IOPPROC].

B.1 SCR for BCAST Service Guide Client (SG-C)

Item	Function	Reference	Status	Requirement
BCAST-SG-C-001	Support for access to Interaction Channel	General	O	BCAST-SG-C-003 AND BCAST-SG-C-006 AND BCAST-SG-C-012 AND BCAST-SG-C-014
BCAST-SG-C-002	Support for Service Guide data model and XML Schema for Schedule, Content, Access, PurchaseItem, PurchaseData, PurchaseChannel and PreviewData fragments of Service Guide	Section 5.1	M	
BCAST-SG-C-003	Support for XML Schema for InteractivityData fragment of Service Guide	Section 5.1	O	
BCAST-SG-C-004	Display of all declared program guide elements or attributes in Service Guide fragments for presentation to end-users	Section 5.1.2	O	
BCAST-SG-C-005	Support for reception of preview data over Broadcast Channel	Section 5.1.2.9	M	
BCAST-SG-C-006	Support for reception of preview data over Interaction Channel	Section 5.1.2.9	O	
BCAST-SG-C-007	Support for split-TOI scheme	Section 5.4.1.1	O	
BCAST-SG-C-008	Support Service Guide encapsulation	Section 5.4.1.3	M	
BCAST-SG-C-009	Support GZIP for SGDU and SGDD compression	Section 5.4.1.4	M	
BCAST-SG-C-010	Support Service Guide Delivery Descriptor	Section 5.4.1.5	M	
BCAST-SG-C-011	Service Guide delivery over Broadcast Channel	Section 5.4.2	M	

Item	Function	Reference	Status	Requirement
BCAST-SG-C-012	Service Guide delivery over Interaction Channel	Section 5.4.3	O	
BCAST-SG-C-013	Support Service Guide update and management over Broadcast Channel	Section 5.5.1	M	
BCAST-SG-C-014	Support Service Guide update and management over Interaction Channel	Section 5.5.2	O	

B.2 SCR for BCAST Service Guide Generation/Adaptation/Distribution (SG-GAD)

Item	Function	Reference	Status	Requirement
BCAST-SGGAD-S-001	SGGAD-S serving BDS that is capable of supporting Broadcast Channel	General	O	BCAST-SGGAD-S-019 AND BCAST-SGGAD-S-021 AND BCAST-SGGAD-S-023
BCAST-SGGAD-S-002	SGGAD-S serving BDS that is capable of supporting Interaction Channel	General	O	BCAST-SGGAD-S-020 AND BCAST-SGGAD-S-022 AND BCAST-SGGAD-S-024
BCAST-SGGAD-S-003	Backend interface SG-2 exposed in implementation	Section 5.6	O	BCAST-SGGAD-S-004
BCAST-SGGAD-S-004	Support backend interface SG-2 for SG function	Section 5.3.1	O	BCAST-SGGAD-S-005
BCAST-SGGAD-S-005	Support Service Guide data model and the corresponding XML schema for the Service Guide data exchange	Section 5.1.2 , Section 5.3.1	O	
BCAST-SGGAD-S-006	Backend interface SG-4 exposed in implementation	Section 5.6	O	BCAST-SGGAD-S-007
BCAST-SGGAD-S-007	Support backend interface SG-4 for SG function	Section 5.6	O	(BCAST-SGGAD-S-008 OR BCAST-SGGAD-S-009) AND BCAST-SGGAD-S-010 AND (BCAST-SGGAD-S-011 OR BCAST-SGGAD-S-012) AND BCAST-SGGAD-S-013 AND BCAST-SGGAD-S-005
BCAST-SGGAD-S-008	Support IPv4	Section 5.6.2.1	O	
BCAST-SGGAD-S-009	Support IPv6	Section 5.6.2.1	O	
BCAST-SGGAD-S-010	Support TCP	Section 5.6.2.1	O	
BCAST-SGGAD-S-011	Support HTTP1.1	Section 5.6.2.1	O	

Item	Function	Reference	Status	Requirement
BCAST-SGGAD-S-012	Support HTTPS	Section 5.6.2.1	O	
BCAST-SGGAD-S-013	SG backend messages for content delivery	Section 5.6.2.2	O	
BCAST-SGGAD-S-014	Support announcing Service Guides within a service guide	Section 5.2	O	
BCAST-SGGAD-S-015	Support Service Guide fragment identification	Section 5.4.1.1	M	
BCAST-SGGAD-S-016	Support Service Guide encapsulation	Section 5.4.1.3	M	
BCAST-SGGAD-S-017	Support GZIP for SGDU and SGDD compression	Section 5.4.1.4	O	
BCAST-SGGAD-S-018	Support Service Guide Delivery Descriptor	Section 5.4.1.5	M	
BCAST-SGGAD-S-019	Service Guide delivery over Broadcast Channel	Section 5.4.2	O	
BCAST-SGGAD-S-020	Service Guide delivery over Interaction Channel	Section 5.4.3	O	
BCAST-SGGAD-S-021	Support Service Guide update and management over Broadcast Channel	Section 5.5.1	O	
BCAST-SGGAD-S-022	Support Service Guide update and management over Interaction Channel	Section 5.5.2	O	
BCAST-SGGAD-S-023	Support delivery of preview data over Broadcast Channel	Section 5.1.2.9	O	
BCAST-SGGAD-S-024	Support delivery of preview data over Interaction Channel	Section 5.1.2.9	O	

B.3 SCR for BCAST Service Guide Subscription Source (SGSS)

Item	Function	Reference	Status	Requirement
BCAST-SGSS-S-001	Backend interface SG-4 exposed in implementation	Section 5.6	O	BCAST-SGSS-S-002
BCAST-SGSS-S-002	Support backend interface SG-4 for SG function	Section 5.6	O	(BCAST-SGSS-S-003 OR BCAST-SGSS-S-004) AND BCAST-SGSS-S-005 AND (BCAST-SGSS-S-006 OR BCAST-SGSS-S-007) AND BCAST-SGSS-S-008 AND BCAST-SGSS-S-009
BCAST-SGSS-S-003	Support IPv4	Section 5.6.2.1	O	
BCAST-SGSS-S-004	Support IPv6	Section 5.6.2.1	O	
BCAST-SGSS-S-005	Support TCP	Section 5.6.2.1	O	

Item	Function	Reference	Status	Requirement
BCAST-SGSS-S-006	Support HTTP1.1	Section 5.6.2.1	O	
BCAST-SGSS-S-007	Support HTTPS	Section 5.6.2.1	O	
BCAST-SGSS-S-008	SG backend message for content delivery	Section 5.6.2.2	O	
BCAST-SGSS-S-009	Support Service Guide data model and the corresponding XML schema for the Service Guide data exchange	Section 5.3.1	O	

B.4 SCR for BCAST Service Guide Application Source (SGAS)

Item	Function	Reference	Status	Requirement
BCAST-SGAS-S-001	Backend interface SG-1 exposed in implementation	Section 5.6	O	BCAST-SGAS-S-002
BCAST-SGAS-S-002	Support backend interface SG-1 for SG function	Section 5.3.1	O	
BCAST-SGAS-S-003	Backend interface SG-2 exposed in implementation	Section 5.6	O	BCAST-SGAS-S-004
BCAST-SGAS-S-004	Support backend interface SG-2 for SG function	Section 5.3.1	O	BCAST-SGAS-S-005
BCAST-SGAS-S-005	Support Service Guide data model and the corresponding XML schema for the Service Guide data exchange	Section 5.3.1	O	

Appendix C. Service Guide Application Scenarios (Informative)

C.1 Scenario of selective viewing based on the codec requirement in the SG

This application scenario describes how the same content using different codec can be subscribed and consumed by the terminals with different AV Codec capabilities via different channels.

Assuming there is a video streaming program in two different formats, MPEG-2 and H.264. To facilitate the terminals that support H.264 to access the high quality video stream, and the terminals that only support MPEG-2 to access the low quality video stream, the following steps may be used.

Step1. SG provider creates one ‘Service’ fragment for the video streaming program that is encoded in two different formats MPEG-2 and H.264.

Step 2. SG provider creates two ‘Access’ fragments associated with the above ‘Service’ fragment to respectively specify the two access channels for the two video stream formats. In the "TerminalCapabilityRequirement" element of every ‘Access’ fragment, the video codec support requirement is explicitly specified.

Step 3. After receiving and reading the Service Guide, the terminal or the user selects the preferred access channel based on the terminal’s AV decoding capabilities. Then the terminal accesses to the selected channel which delivers the corresponding video stream. Please note, if the service is not free, some necessary subscription procedures will be involved.

C.2 Scenario of selective viewing based on the bandwidth requirement in the SG

This application scenario describes how the SG can support the selective viewing of the different versions (for example, basic version only contains audio; normal version contains both audio and video; or the basic version contains the low bit rate stream of the live show, but the normal version contains the high bit rate stream of the same live show) of the same real-time program with different bandwidth requirements. The selective viewing provides more flexibility to the terminal users and ensures the users can always consume their interested program even as the terminal is under a bad reception condition, consequently enhances the user experience.

Assuming there are two versions of the same streaming program, the basic version requires 200kbps for the bandwidth and the normal version requires 500kbps as the bandwidth. To facilitate the terminals that are under good reception condition (>500kbps) to access the normal version of the program, and the terminals that are under the bad reception condition (<200kbps) to access the basic version of the program, the following steps may be used.

Step1. SG provider creates a ‘Service’ fragment for the streaming program.

Step 2. SG provider creates two ‘Access’ fragments associated with the above ‘Service’ fragment to respectively specify the two access channels, one access only delivers the basic version of the live show which only contains the audio component or contains the low bit rate streams of the original audio and video streams, another access delivers the normal version of the live show which contains the original high bit rate streams of the audio and video streams. In the "BandwidthRequirement" element of every ‘Access’ fragment, the bandwidth requirement for every access is explicitly specified.

Step 3. As the program is broadcast via two access channels, the terminal or the user selects the appropriate access channel based on the current reception condition of the terminal. If the terminal reception condition can not meet the bandwidth requirement of the normal version of the access, then the terminal will listen to basic access channel. Otherwise, the terminal will be able to access to the normal access channel.

C.3 Scenario of a TV program which is broadcasted in two languages.

In this scenario a TV show on a certain TV channel is broadcasted in two different languages.

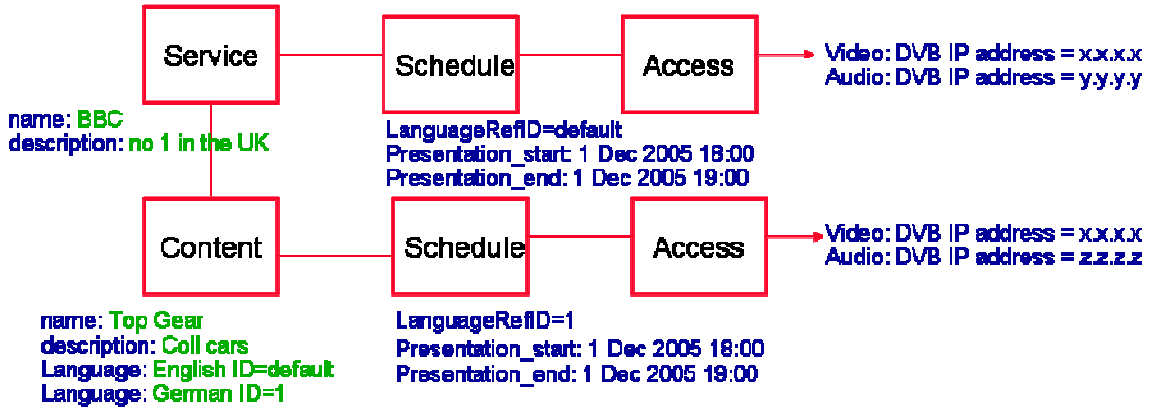


Figure 14: Use of service, content, schedule and 'Access' fragment to declare a service with a TV program in different languages.

The default language of the show "Top Gear" is in English. However, there is a second language available for this program, German. The default language inherits the access parameters from the 'Service' fragment it is related to. The second language declares a different source IP address for the audio stream.

This is not only useful for TV programs with a second language, but also for roaming users. They might be provided with an 'Access' fragment pointing to a UMTS stream, which gives them the audio comment in their local language via the interactive link. These ESG fragments for roaming users could be either distributed via the back-end systems over the broadcast bearer or retrieved via the interactive link.

C.4 Scenario of a broadcast streaming service protected by multiple Key Management Systems

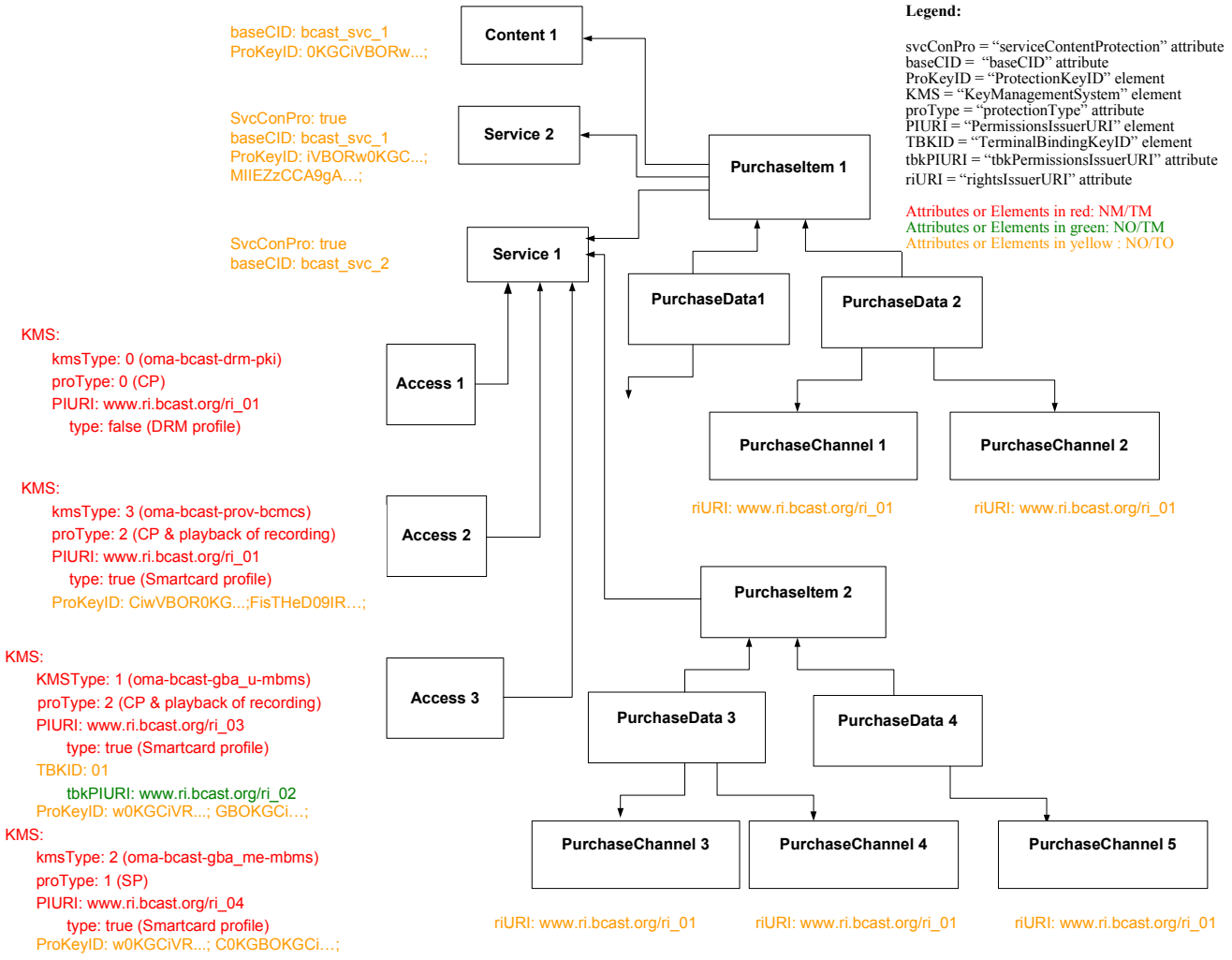


Figure 14: Use of SG to signal service and content protection information

This application scenario describes how the service and content protection information of a broadcast service protected by multiple Key Management Systems (KMS) is signalled using the Service Guide.

First of all, to generally inform the terminal and the user whether a service is protected or not the 'serviceContentProtection' attribute in the associated 'Service' fragment is set to "true" or "false".

In the example shown in Figure 15, to access the protected Service 1 there are three accesses specified by three 'Access' fragments, Access 1, Access 2 and Access 3, for each of which different KMS are used. The KMS related information is signalled by 'KeyManagementSystem' element in the 'Access' fragment. From the linkage between the 'Service' fragment and the 'Access' fragments, the terminal can easily find out by how many and what exact KMSs a specific service is protected.

In this example, the KMS type of Access 1 is "oma-bcast-drm-pki", the protection type used with this KMS is Content Protection as defined by LTKM (i.e. protection_after_reception in STKM = 0x00 or 0x01 [BCAST10-ServContProt]); the Permissions Issuer URI to acquire SEK/PEK is "www.ri.bcast.org/ri_01", and in case of Access 1 the Permissions Issuer is synonymous with the Rights Issuer because DRM Profile is used here.

The KMS type of Access 2 is “oma-bcast-prov-bcmcs”, the protection type used with this KMS is Content Protection as defined by LTKM, plus playback of protected recording permission (i.e. protection_after_reception in STKM = 0x02); The Permissions Issuer URI to acquire SEK/PEK is also “www.ri.bcast.org/ri_01”, identical to that for Access 1, and which means one Permissions Issuer can support multiple KMSs.

For Access 3 two types of KMS are supported; one is “oma-bcast-gba_u-mbms”, the other is “oma-bcast-gba_me-mbms”. The protection type of the former is Content Protection as defined by LTKM, plus playback of protected recording permission (i.e. protection_after_reception in STKM = 0x02). Protection type of the latter KMS is Service Protection only (i.e. protection_after_reception in STKM = 0x03). The Permissions Issuer URIs are “www.ri.bcast.org/ri_03” and “www.ri.bcast.org/ri_04” respectively.

Besides the above service and content information which are common to both DRM profile and Smartcard profile, some other parameters in SG are only specific to one profile. Those parameters are only required to be supported when the relevant profile is supported.

For the DRM Profile, part of the Service or Program CID used to identify the corresponding asset within a OMA DRM 2.0 Rights Object are signalled as ‘baseCID’ attribute in the corresponding ‘Service’ fragment and ‘Content’ fragment. The Service or Program CID is obtained from the BaseCID as described in [BCAST10-ServContProt]. Furthermore, to inform the terminal which Rights Issuer may provide the Rights Issuer Services [DRM20-Broadcast-Extensions] operated by its home BSM, a ‘rightsIsserURI’ attribute may also be signalled in the ‘PurchaseChannel’ fragment in case of DRM profile.

For the Smartcard profile, the list of the key identifiers needed to access the protected service or content are signalled as ‘ProtectionKeyID’ element in ‘Service’, ‘Content’, ‘Access’ or ‘PurchaseItem’ fragment so that the terminal can determine whether or not it has the correct key material to access the service or content. In the above example, the key identifiers for Content 1 and Service 2 are indicated in the fragment itself because they are not shared by multiple service providers. However, for Service 1, because multiple means of access, associated with different KMS(s) are provided (possibly each from a different service provider), the protection keys in each KMS are different. In this case the protection key identifiers are signalled under ‘KeyManagementSystem’ element of each ‘Access’ fragment, rather than in the ‘Service’ fragment. Furthermore, in the case of Smartcard Profile, Terminal Binding Key (TBK) can be used to bind the broadcast service to valid terminals for content protection. The identifier of the TBK is signalled as ‘TerminalBindingKeyID’ sub-element of ‘KeyManagementSystem’. The GBA_U KMS of Access 3 in the above example uses TBK. Because the Permissions Issuer from which the TBK is acquired may be different than the Permissions Issuer from which the SEK/PEK is acquired, a TBK specific Permissions Issuer is specified as ‘tbkPermissionsIssuerURI’ attribute under ‘TerminalBindingKeyID’ element. However in some cases the same Permissions Issuer is used to provide both SEK/PEK and TBK, and for which the ‘tbkPermissionsIssuerURI’ attribute will not be instantiated.

C.5 Scenario of a main service with interaction

This application scenario describes how the Service Guide can be used to describe the scenario in which the main service (for example, mobile TV) is enhanced with interaction (for example, SMS voting). Figure 16 depicts the scenario on a timeline. Figure 17 depicts the necessary Service Guide fragment and their relation. Figure 18 depicts the example encapsulation to SGDU. Note: in this example case the schedule of content is the same as the delivery schedule of InteractivityMediaDocuments. If different schedule would be needed, can one achieve that by instantiating two 'Schedule' fragments – one for Content and one for the delivery of Interactive Media Documents.

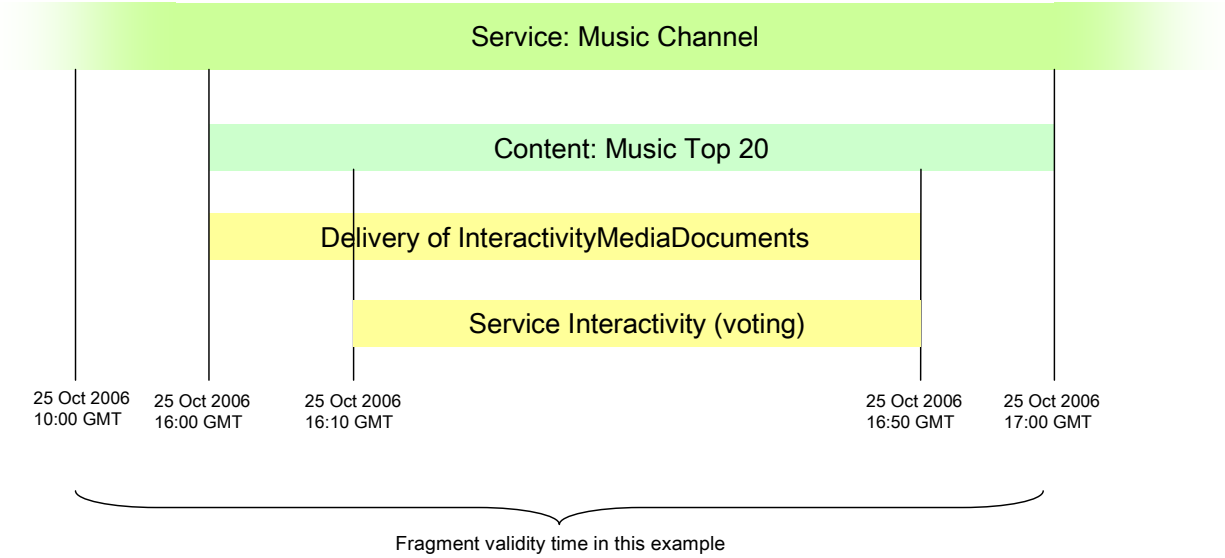


Figure 156: Service example on a timeline

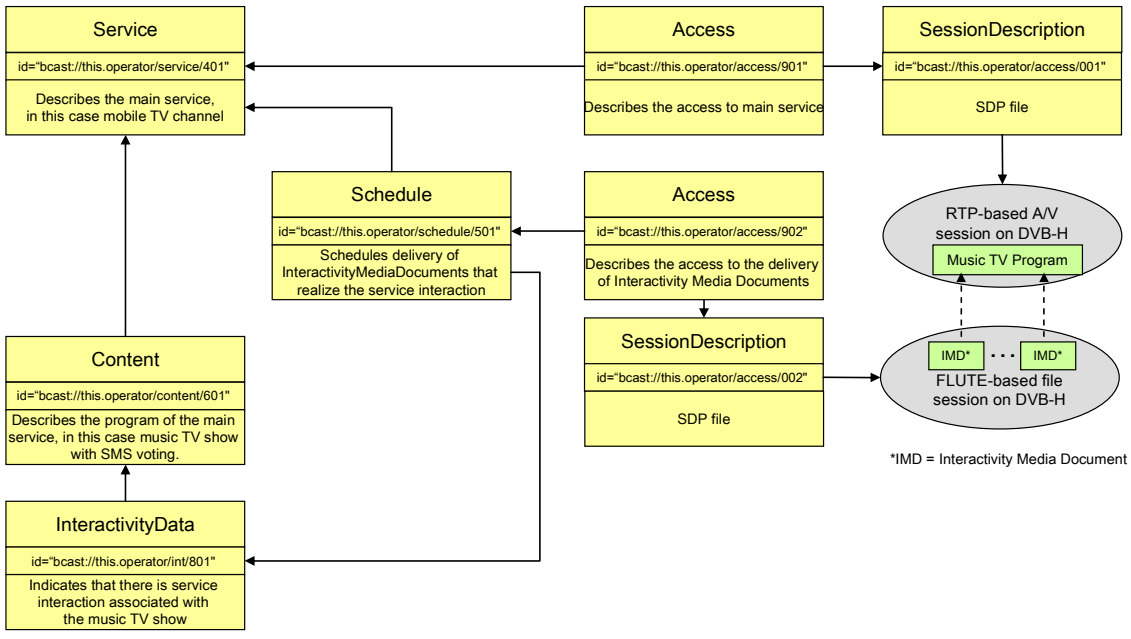


Figure 167: Service Guide example for main service with interactivity

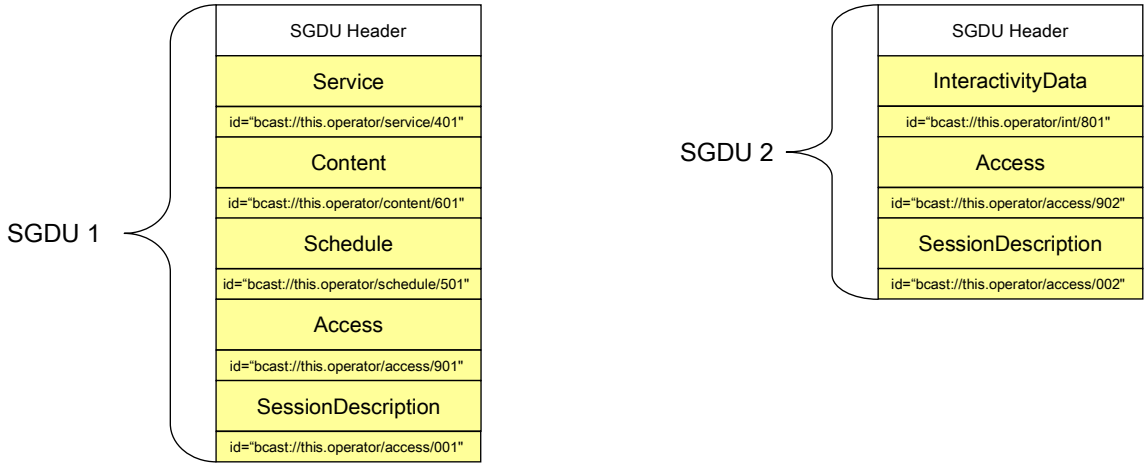


Figure 178: Simple encapsulation of Service Guide fragments in this example

The following sections give example instantiations of the Service Guide fragments in this example.

C.5.1 Service (bcast://this.operator/service/401)

Name	Type	Example	Data Type
Service	E		
id	A	bcast://this.operator/service/401	anyURI
version	A	1	unsignedInt
validFrom	A	25 Oct 2006, 10:00 GMT in NTP	unsignedInt
validTo	A	25 Oct 2006, 17:00 GMT in NTP	unsignedInt
weight	A	2	unsignedInt
serviceContentProtection	A	false	boolean
ServiceType	E1	1 (Basic TV)	unsigned Byte
		Start of program guide	
Name	E1	Music Channel (xml:lang="eng")	string
Name	E1	Musiikkikanava (xml:lang="fin")	string
Genre	E1	Music	string
Extension	E1		
url	A	http;//this.operator/music_channel.html	anyURI
Description	E2	Home page of this service (xml:lang="eng")	string
Description	E2	Kanavan kotisivu (xml:lang="fin")	string
		End of program guide	

C.5.2 Schedule (bcast://this.operator/schedule/501)

Name	Type	Example	Data Type
Schedule	E		
id	A	bcast://this.operator/schedule/501	anyURI
version	A	5	unsignedInt
defaultSchedule	A	true	boolean
validFrom	A	25 Oct 2006, 10:00 GMT in NTP	unsignedInt
validTo	A	25 Oct 2006, 17:00 GMT in NTP	unsignedInt
ServiceReference	E1		
idRef	A	bcast://this.operator/service/401	anyURI
InteractivityDataReference	E1		

idRef	A	bcast://this.operator/int/801	anyURI
DistributionWindow	E2		
startTime	A	25 Oct 2006, 16:00 GMT in NTP	unsignedInt
endTime	A	25 Oct 2006, 16:50 GMT in NTP	unsignedInt

C.5.3 Content (bcast://this.operator/content/601)

Name	Type	Example	Data Type
Content	E		
id	A	bcast://this.operator/content/601	anyURI
version	A	10	unsignedInt
validFrom	A	25 Oct 2006, 10:00 GMT in NTP	unsignedInt
validTo	A	25 Oct 2006, 17:00 GMT in NTP	unsignedInt
Type	A	“video”	string
ServiceReference	E1		
idRef	A	bcast://this.operator/service/401	anyURI
		Start of program guide	
Name	E1	Music Top 20 (xml:lang="eng")	String
StartTime	E1	2006-10-25T16:00:00Z	dateTime
EndTime	E1	2006-10-25T17:00:00Z	dateTime
Genre	E1	Music	string
		End of program guide	

C.5.4 InteractivityData (bcast://this.operator/int/801)

Name	Type	Example	Data Type
InteractivityData	E		
id	A	bcast://this.operator/int/801	anyURI
version	A	2	unsignedInt
validFrom	A	25 Oct 2006, 10:00 GMT in NTP	unsignedInt
validTo	A	25 Oct 2006, 10:00 GMT in NTP	unsignedInt
preListenIndicator	A	false	boolean
interactivityMediaDocumentPointer	A	1881rt45tqw	anyURI
InteractivityType	E1	"voting"	string
ContentReference	E1		
idRef	A	bcast://this.operator/content/601	anyURI
InteractivityWindow	E1		
startTime	A	25 Oct 2006, 16:10 GMT in NTP	unsignedInt
endTime	A	25 Oct 2006, 16:50 GMT in NTP	unsignedInt
BackOffTiming	E1		
offsetTime	A	3	decimal
randomTime	A	2	decimal

C.5.5 Access (bcast://this.operator/access/901)

Name	Type	Example	Data Type
Access	E		
id	A	bcast://this.operator/access/901	anyURI
version	A	2	unsignedInt
validFrom	A	25 Oct 2006, 10:00 GMT in NTP	unsignedInt
validTo	A	25 Oct 2006, 17:00 GMT in NTP	unsignedInt
AccessType	E1		
BroadcastServiceDelivery	E2		
BDSType	E3	0 (IPDC over DVB-H)	unsignedByte
Version	E4	1	string
SessionDescriptionReference	E3		
type	A	0 – SDP	unsignedByte
idRef	A	bcast://this.operator/access/001	anyURI
ServiceReference	E1		
idRef		bcast://this.operator/service/401	anyURI
ServiceClass	E1	urn:oma:bcast:oma_bsc:st:1.0	string

C.5.6 Access (bcast://this.operator/access/902)

Name	Type	Example	Data Type
Access	E		
id	A	bcast://this.operator/access/902	anyURI
version	A	8	unsignedInt
validFrom	A	25 Oct 2006, 10:00 GMT in NTP	unsignedInt
validTo	A	25 Oct 2006, 17:00 GMT in NTP	unsignedInt
AccessType	E1		
BroadcastServiceDelivery	E2		
BDSType	E3	0. IPDC over DVB-H	unsignedByte
Version	E4	1	string
SessionDescriptionReference	E3		
type	A	0 – SDP	unsignedByte
idRef	A	bcast://this.operator/access/002	anyURI
ScheduleReference	E1		
idRef	A	bcast://this.operator/schedule/502	anyURI
ServiceClass	E1	urn:oma:bcast:oma_bsc:fc:1.0	String

C.6 Scenario of a main service with purchasable protected auxiliary data content

This application scenario describes how the Service Guide can be used to describe the scenario in which the main service (for example, mobile TV) is enhanced with purchasable, protected auxiliary data (for example, sports statistics related to sports event main service). In this case, the main service is provided free to air. Figure 19 depicts the scenario on a timeline. Figure 20 depicts the necessary Service Guide fragments and their relations. Figure 21 depicts the example encapsulation to SGDU.

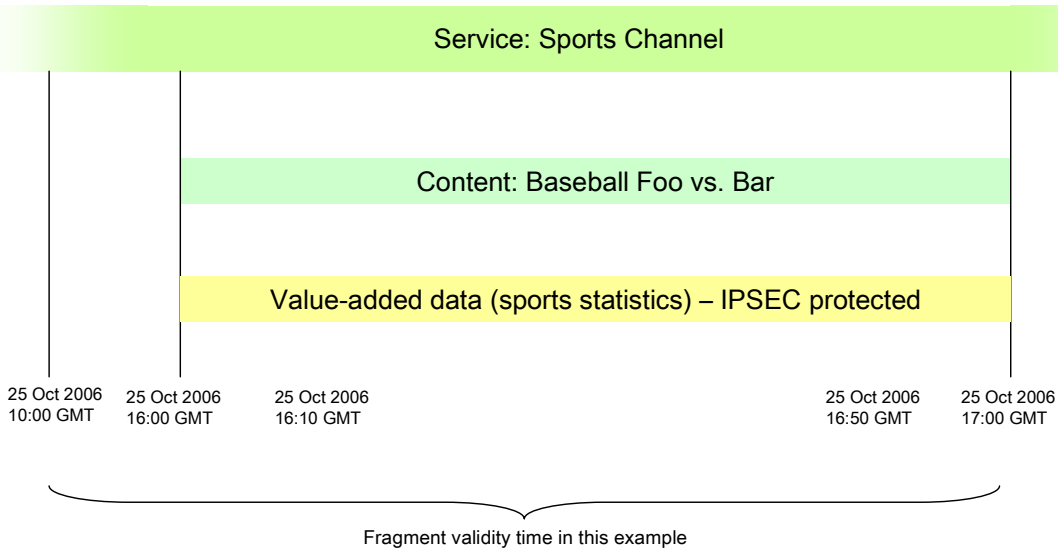


Figure 189: Service example on a timeline

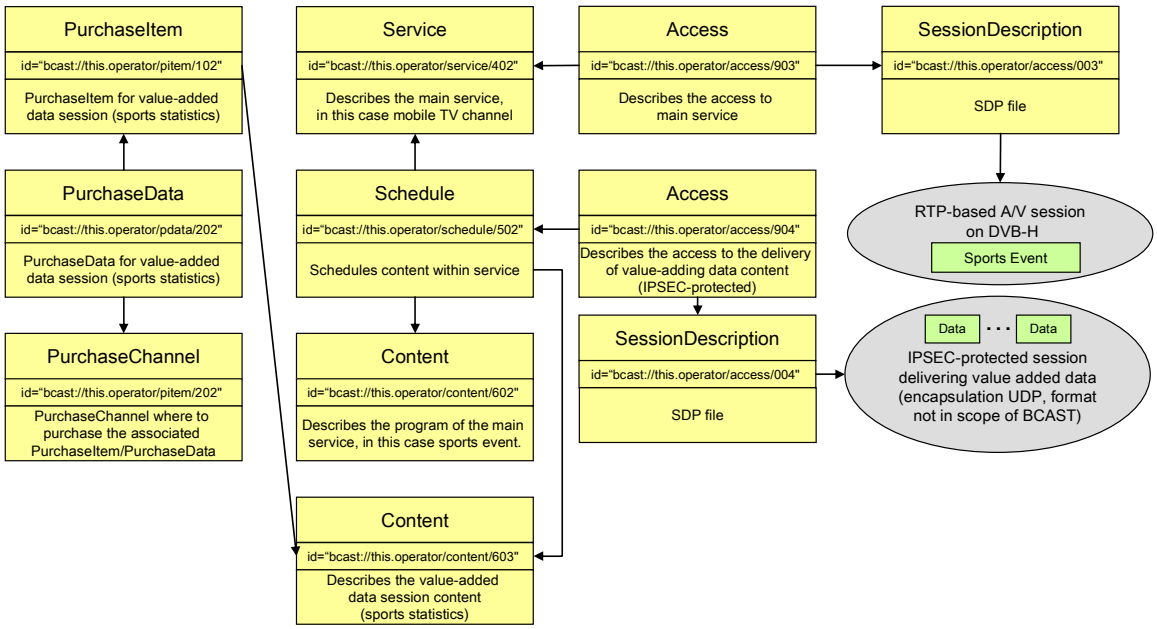


Figure 20: Service Guide example for main Service with value-added, protected data

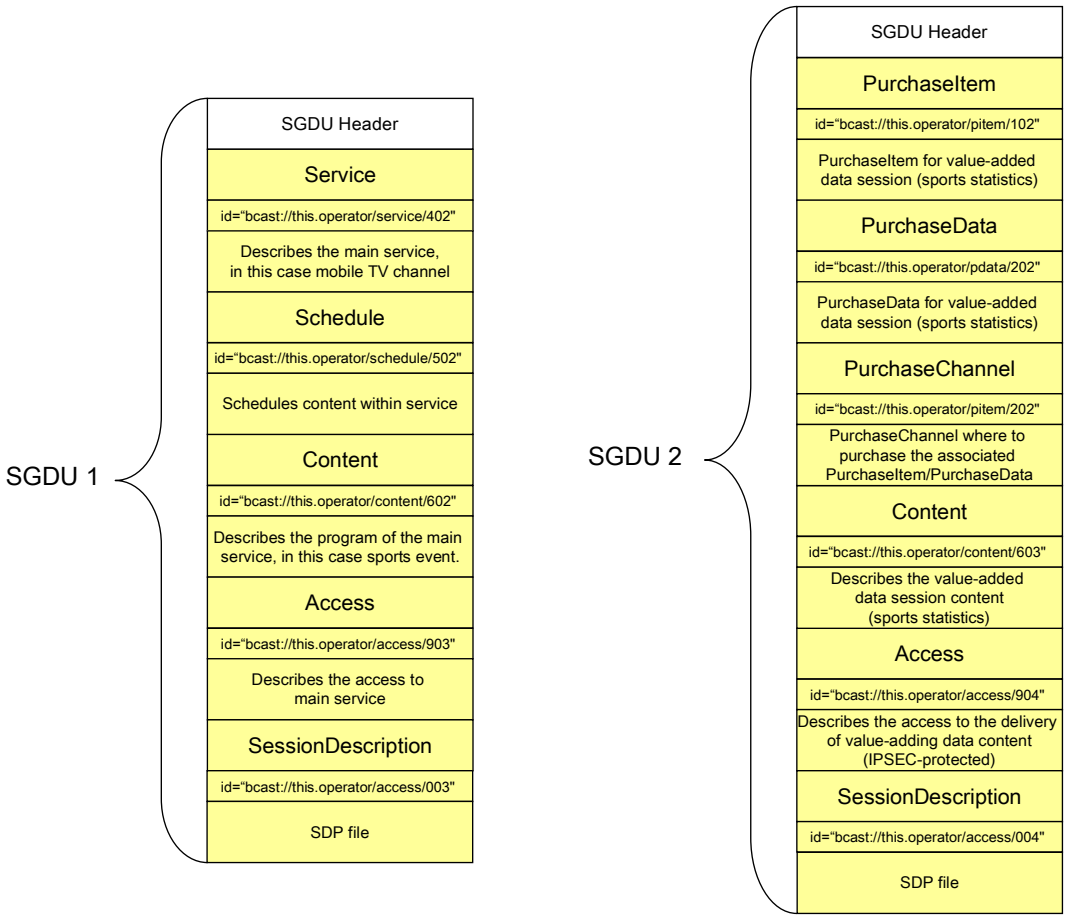


Figure 21: Simple encapsulation of Service Guide fragments in this example

C.7 Scenario of multiple services provided with single channel each

This application scenario describes how the Service Guide can be used to describe the scenario in which multiple services provided with single channel each. Figure 22 depicts the necessary Service Guide fragments and their relations. Figure 23 depicts the example encapsulation to SGDUs.

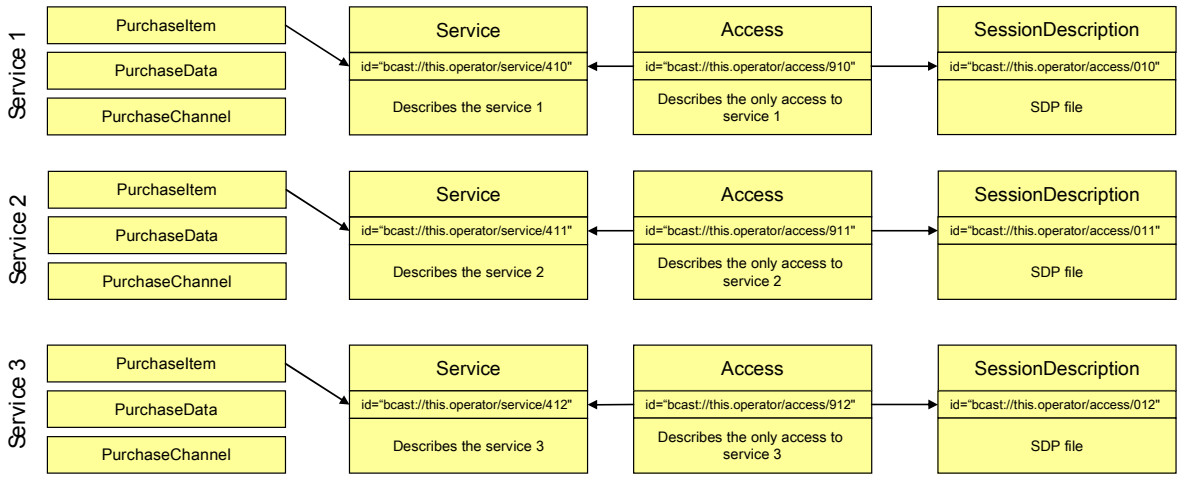


Figure 192: Service Guide example

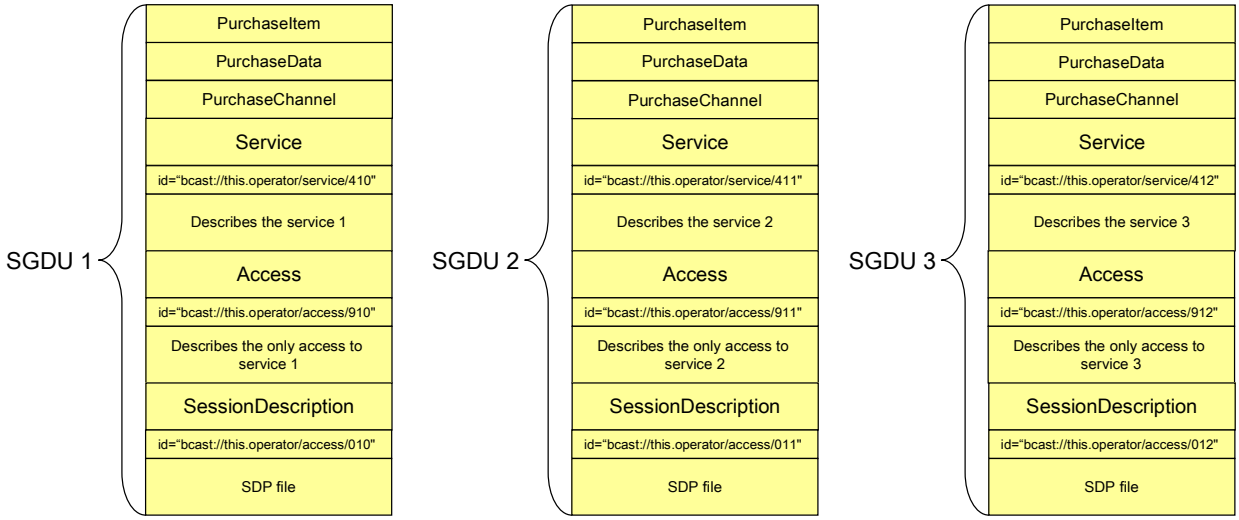
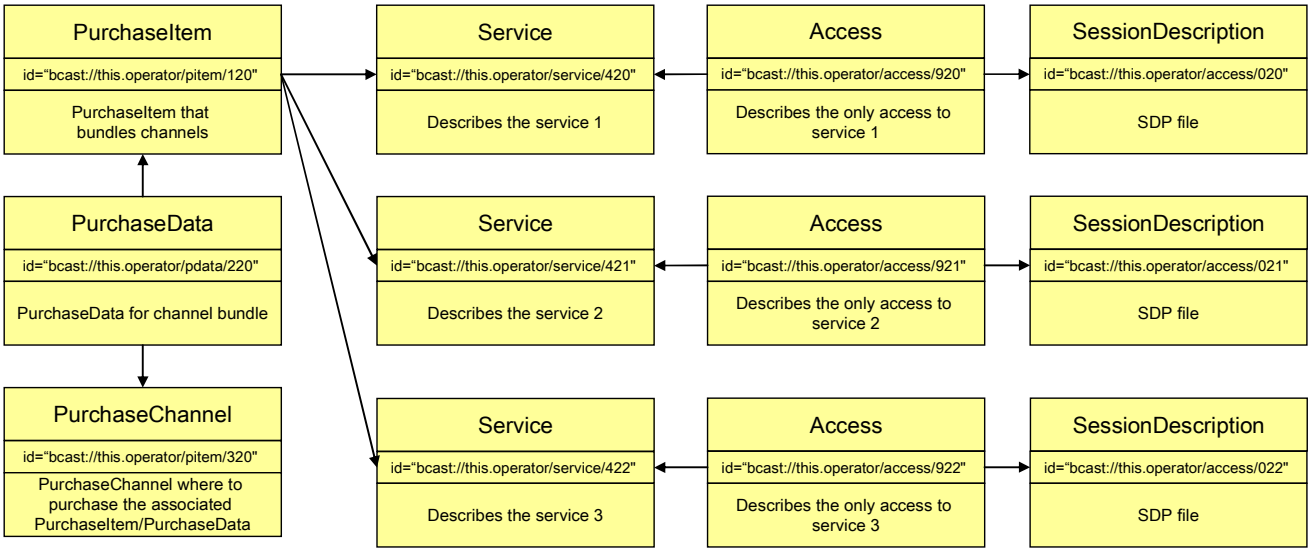


Figure 203: Simple encapsulation of Service Guide fragments in this example

C.8 Scenario of bundling multiple channels into a single service

This application scenario describes how the Service Guide can be used to describe the scenario in which multiple channels are bundled with single service. In practice, each channel is represented as service and bundling is done through PurchaseItem. Figure 24 depicts the necessary Service Guide fragments and their relations. Figure 24 depicts the example encapsulation to SGDUs.



. . .

Figure 214: Service Guide figure

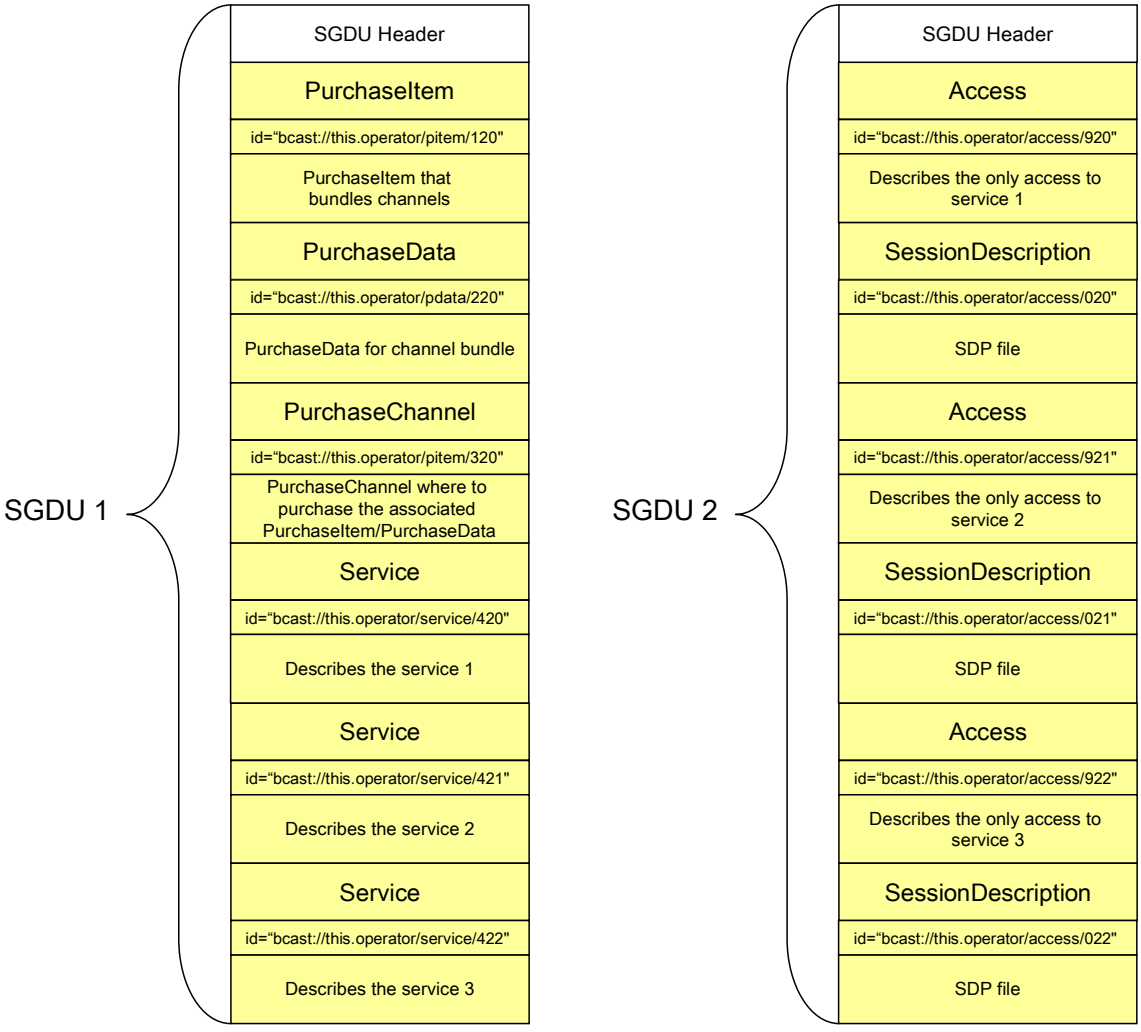


Figure 225: Simple encapsulation of Service Guide fragments in this example

C.9 Scenario of using single Service Guide to provide service descriptions for multiple service providers

This application scenario describes how single Service Guide provides separate service descriptions for multiple service providers. In this case the main Service Guide is an aggregate of individual service descriptions, each associated with an individual Service Provider. Figure 26 depicts the necessary Service Guide fragments and their relations. In this example, two Service Providers (X and Y) are providing Service Guide fragments within the same Service Guide.

The association between the service providers and the individual fragments is provided using the grouping method of SGDD. The example grouping applying to this example case is provided in Tables 6 and 7. The eight fragments of the example are thought to be delivered using two SGDUs having TOIs two and nine. The SGDU having TOI two contains the fragments associated with the operator X and the SGDU having TOI nine contains the fragments associated with operator Y. The exception is 'Service' fragment having id bcast://this.operator/service/431. That fragment is associated with both the operators X and Y being delivered in the SGDU having TOI two. These eight fragments are in turn declared using two SGDDs, one representing the fragments associated with the operator X and the other representing the fragments associated with operator Y. Both the SGDDs also represent the subset of the Service Guide representing information for time interval from one to two P.M. on the 20th of December 2006. This is achieved using two grouping criteria, namely the time grouping criteria and BSM selector criteria, simultaneously.

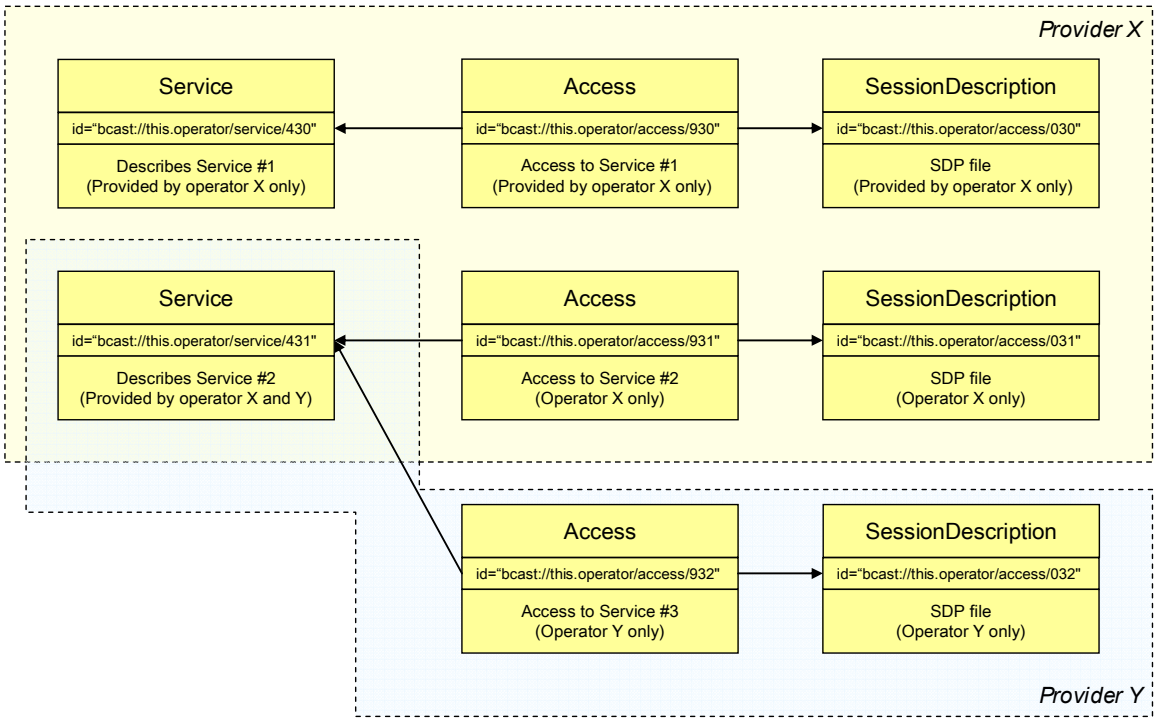


Figure 236: Service Guide figure

Name	Type	Example	Data Type
ServiceGuideDeliveryDescriptor	E		
id	A	bcast://bsda.com/sgdd/1234	anyURI
version	A	3	unsignedInt
NotificationReception	E1		
port	A	34343	unsignedInt

address	A	232.0.0.1	string
DescriptorEntry	E1		
GroupingCriteria	E2		
TimeGroupingCriteria	E3		
startTime	A	20 Dec 2006, 13:00 GMT in NTP	unsignedInt
endTime	A	20 Dec 2006, 14:00 GMT in NTP	unsignedInt
BSMSelector	E3		
id	A	bcast://operatorX.com	anyURI
BSMFilterCode	E4	OPERATOR_X	string
type	A	1	unsignedByte
Name	E4	Operator X	string
Transport	E2		
ipAddress	A	232.0.0.2	string
port	A	8765	unsignedShort
srcIpAddress	A	1.2.3.4	string
transmissionSessionID	A	456	unsignedShort
ServiceGuideDeliveryUnit	E2		
transportObjectID	A	2	unsignedInt
validFrom	A	20 Dec 2006, 13:30 GMT in NTP	
validTo	A	20 Dec 2006, 14:00 GMT in NTP	
Fragment	E3		
transportID	A	1001	unsignedInt
id	A	bcast://this.operator/service/430	anyURI
version	A	4	unsignedInt
Fragment	E3		
transportID	A	1002	unsignedInt
id	A	bcast://this.operator/access/930	anyURI
Version	A	2	unsignedInt
Fragment	E3		
transportID	A	1003	unsignedInt
id	A	bcast://this.operator/access/030	anyURI
Version	A	6	unsignedInt
Fragment	E3		
transportID	A	1004	unsignedInt
id	A	bcast://this.operator/service/431	anyURI
version	A	2	unsignedInt
transportID	A	1005	unsignedInt
id	A	bcast://this.operator/access/931	anyURI
version	A	1	unsignedInt
Fragment	E3		
transportID	A	1006	unsignedInt
id	A	bcast://this.operator/access/031	anyURI
version	A	3	unsignedInt

Table 6: SGDD Example

Name	Type	Example	Data Type
ServiceGuideDeliveryDescriptor	E		
id	A	bcast://bsda.com/sgdd/4567	anyURI
version	A	2	unsignedInt
NotificationReception	E1		
port	A	34343	unsignedInt
address	A	232.0.0.1	string
DescriptorEntry	E1		
GroupingCriteria	E2		
TimeGroupingCriteria	E3		
startTime	A	20 Dec 2006, 13:00 GMT in NTP	unsignedInt
endTime	A	20 Dec 2006, 14:00 GMT in NTP	unsignedInt
BSMSelector	E3		
id	A	bcast://operatorY.com	anyURI
BSMFilterCode	E4	OPERATOR_Y	string
type	A	1	unsignedByte
Name	E4	Operator Y	string
Transport	E2		
ipAddress	A	232.0.0.2	string
port	A	8765	unsignedShort
srcIpAddress	A	1.2.3.4	string
transmissionSessionID	A	456	unsignedShort
ServiceGuideDeliveryUnit	E2		
transportObjectID	A	2	unsignedInt
validFrom	A	20 Dec 2006, 13:30 GMT in NTP	
validTo	A	20 Dec 2006, 14:00 GMT in NTP	
Fragment	E3		
transported	A	1004	unsignedInt
id	A	bcast://this.operator/service/431	anyURI
version	A	2	unsignedInt
ServiceGuideDeliveryUnit	E2		
transportObjectID	A	9	unsignedInt
validFrom	A	20 Dec 2006, 13:30 GMT in NTP	
validTo	A	20 Dec 2006, 14:00 GMT in NTP	
Fragment	E3		
transportID	A	1007	unsignedInt
id	A	bcast://this.operator/access/932	anyURI
version	A	9	unsignedInt
Fragment	E3		
transportID	A	1008	unsignedInt

id	A	bcast://this.operator/access/032	anyURI
version	A	1	unsignedInt

Table 7: SGDD Example

C.10 Scenario of Service Guide announcing other Service Guides

This application scenario describes how single Service Guide can refer to other Service Guides (for example, to provide Service Guides for multiple service providers). In this case the main Service Guide is an aggregate of individual Service Guides that are referred from the top level Service Guide. Provisions of section 5.2 ("Announcing Service Guides within a Service Guide") can be applied. Figure 27 depicts the necessary Service Guide fragments and their relations. Figure 28 depicts the example encapsulation to SGDUs.

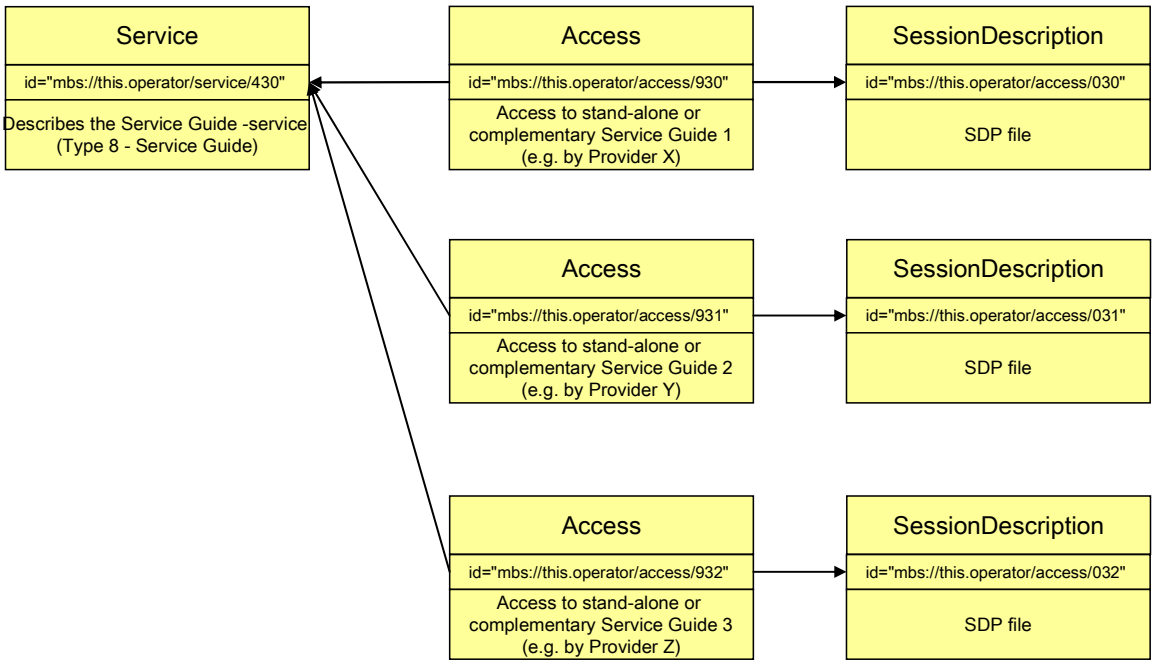


Figure 247: Service Guide example

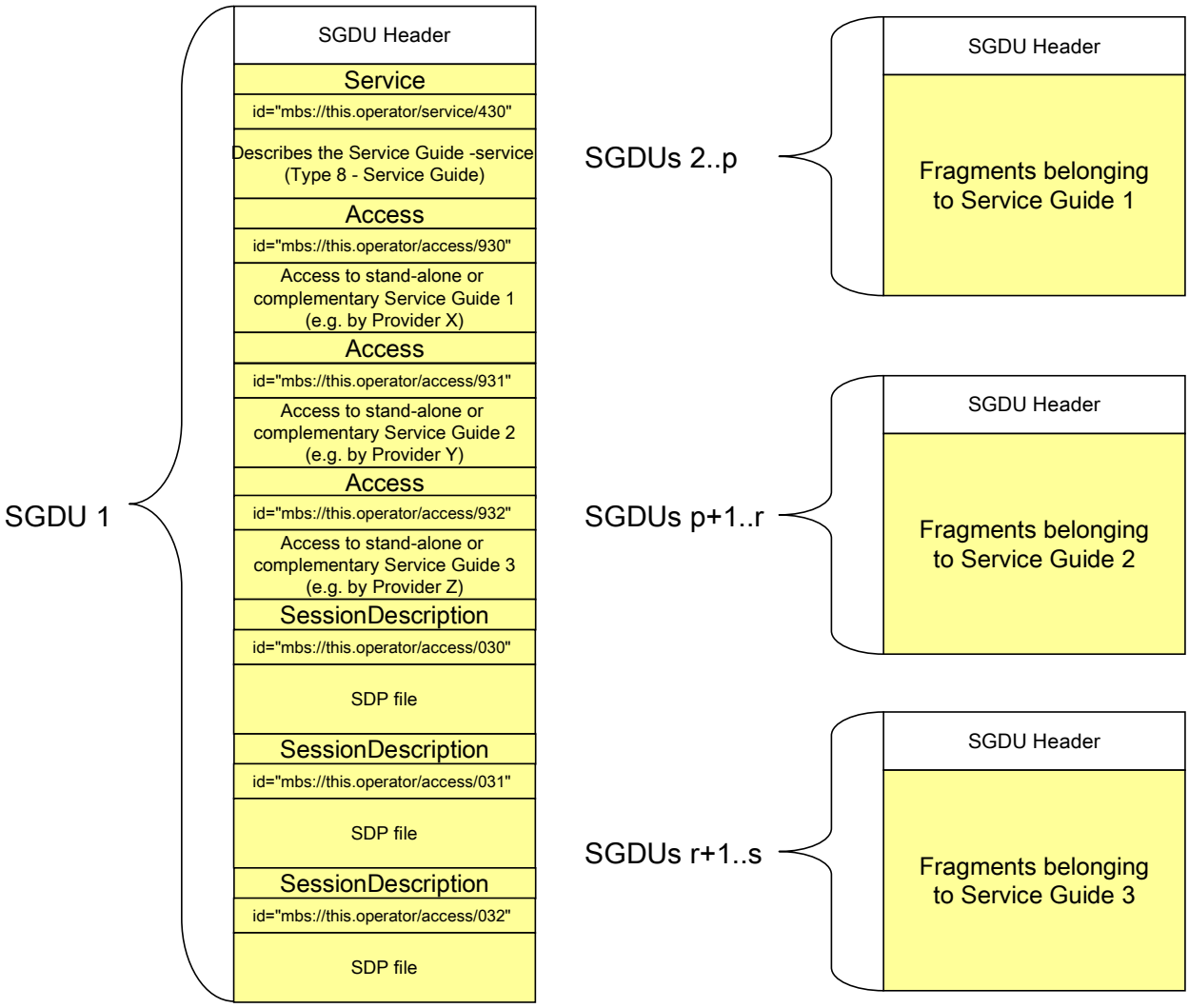


Figure 258: Simple encapsulation of Service Guide fragments in this example

C.11 Scenario of Service Guide with both Broadcast and Unicast Services

A common use of Service Guide is to enable the discovery of both broadcast and unicast services and content within a single place. The application scenario in this section explains how the Service Guide can be instantiated to support such use. In this application scenario, broadcast services are provided over the Broadcast Channel and are live, unidirectional, multicast streaming services. The unicast services are provided over the Interactive Channel and are streaming services available per request, on-demand.

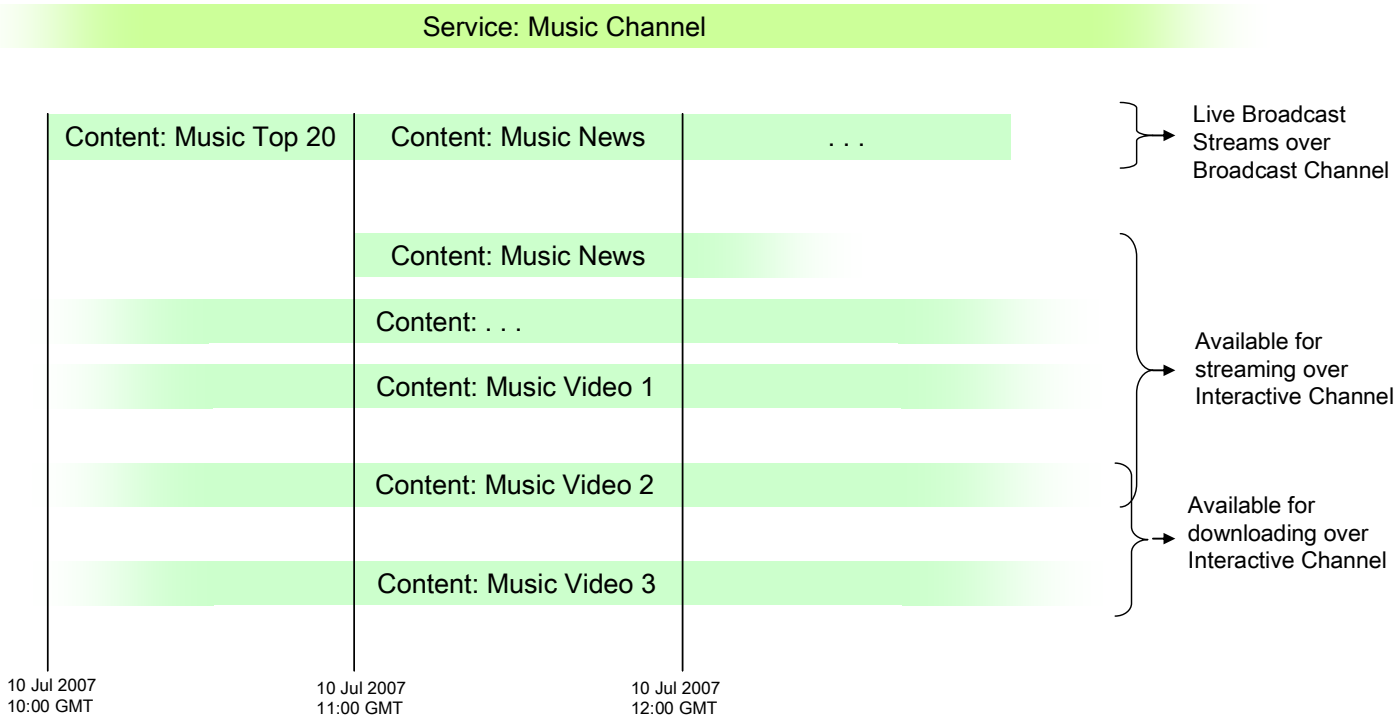


Figure 269: Service example on a timeline

In this example scenario the following is assumed

- There is a service called "Music Channel" that represents a mobile TV / video service delivering music related contents. The users are able to access the "Music Channel" in two ways – either receiving the broadcast streams over Broadcast Channel or accessing the contents over Interactive Channel. These two ways are uniformly represented as a single service on the user interface.
- On the Broadcast Channel the "Music Channel" delivers content so that individual programs such as "Music Top 20" and "Music News" are sequentially scheduled. That means, as the "Music Top 20" ends, it is followed by the next program "Music News".
- Some of the contents associated with service "Music Channel" are available over the Interactive Channel so that the user can select those for streaming. In the example, "Music News", "Music Video 1" and "Music Video 2" are such contents. "Music Video 1" and "Music Video 2" do not have a defined availability time meaning these streams have been available since undefined point of time and will be available until undefined point of time for streaming. "Music News", on the other hand, is scheduled to be available since July 10th, 2007 at 11:00 GMT and remain available until undefined point of time.
- Furthermore, some of the contents associated with service "Music Channel" are available over the Interactive Channel so that the user can select those for downloading. In the example, "Music Video 2" and "Music Video 3" are such contents. Neither of these have defined availability time meaning these contents have been available since undefined point of time and will be available until undefined point of time for delivery as files over Interactive Channel.
- Note that Music News is available over both the Broadcast Channel as well as the Interactive Channel, while "Music Video 1", "Music Video 2" and "Music Video 3" are only available over the interactive channel.
- Contents that are delivered over Broadcast Channel are defined as a purchasable item.
- Contents that are available over Interactive Channel are defined as another purchasable item.

- Note that the same content "Music News" is available on both Broadcast Channel and Interactive Channel at July 10th, 2007 at 11:00 GMT. While at July 10th, 2007 at 12:00 the Broadcast Channel starts to deliver next content, the "Music News" remains available on the Interactive Channel.
- Figure 29 above depicts the scenario on a timeline.

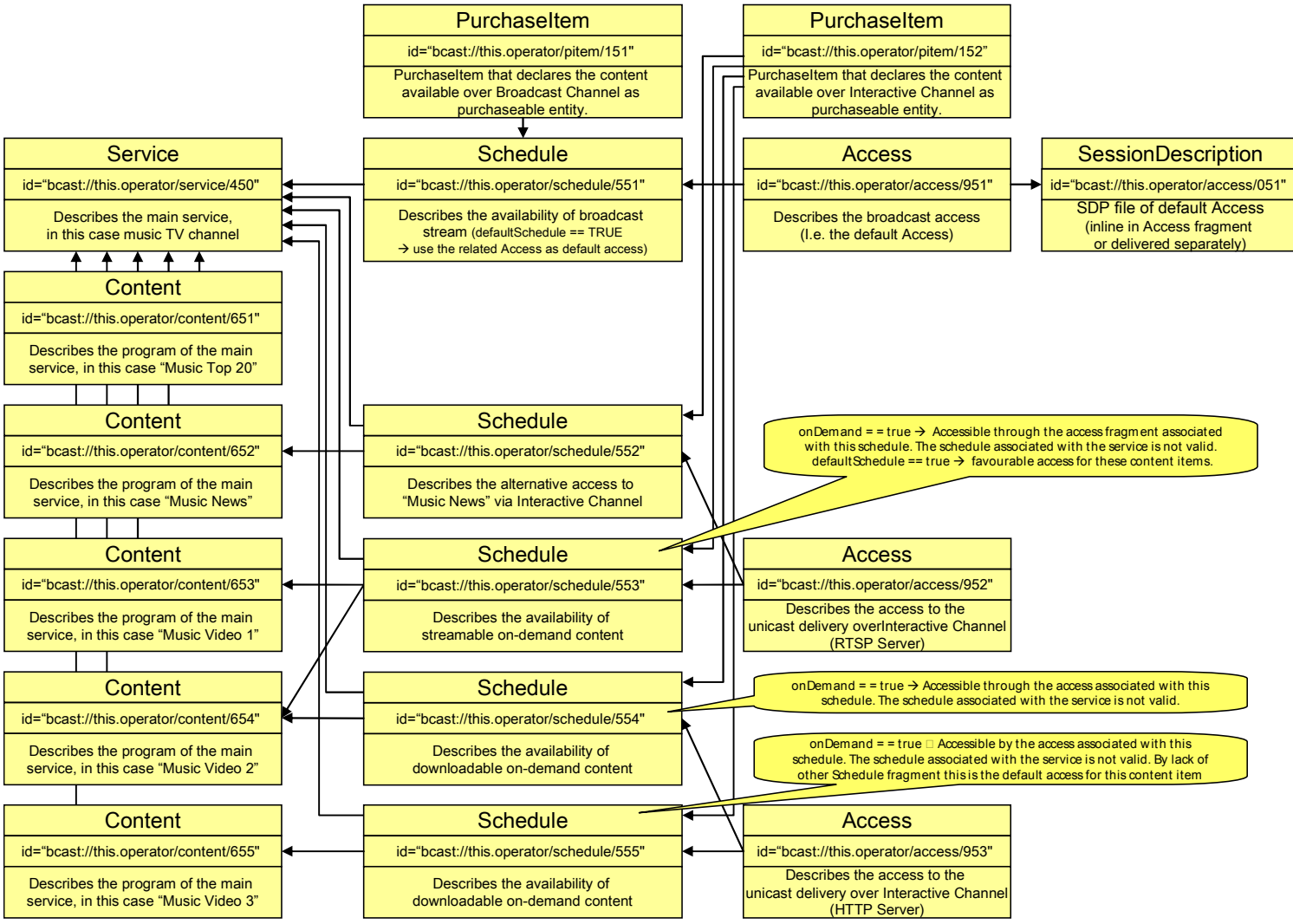


Figure 30: Example of Service Guide instantiation

The Service Guide is instantiated to support the scenario as follows:

- A single 'Service' fragment describes the "Music Channel".
- For each of the contents "Music Top 20", "Music News", "Music Video 1", "Music Video 2" and "Music Video 3" there are an individual 'Content' fragments describing the contents. The 'StartTime' and 'EndTime' within the 'Content' fragments declare the time windows when the corresponding contents are available (either over Broadcast Channel or over Interactive Channel).
- 'Access' fragment with identifier "bcast://this.operator/access/951" describes the access to "Music Channel" over Broadcast Channel. Within the 'Access' fragment the E2-element 'Access.BroadcastServiceDelivery' is instantiated to declare this access is over the Broadcast Channel. The attribute 'Access. BroadcastServiceDelivery BDSType.type' is set to "0 – IPDC over DVB-H" to denote that the delivery takes place over DVB-H bearer.

- 'SessionDescription' is instantiated as SDP description and associated with 'Access' fragment "bcast://this.operator/access/951" and hence describing the way to access the service "Music Channel" over the Broadcast Channel. Alternatively, the 'SessionDescription' could be instantiated as an inline element within the 'Access' fragment.
- Five 'Schedule' fragments are used to describe the availability of contents as follows:
    - The 'Schedule' fragment with identifier "bcast://this.operator/schedule/551" schedules the content associated with "Music Channel" on the Broadcast Channel. This 'Schedule' fragment declares itself being the default 'Schedule' for "Music Channel". This is done by setting the attribute 'defaultSchedule' in the 'Schedule' fragment to 'true'. Consequently this declaration means that the associated 'Access' fragment "bcast://this.operator/access/951" will be the default access for the "Music Channel". Note that this 'Schedule' fragment does not refer to any individual 'Content' fragments.
    - The 'Schedule' fragment with identifier "bcast://this.operator/schedule/552" declares that:
        - the content "Music News" is available for the streaming delivery over the Interactive Channel.
        - this 'Schedule' fragment describes the alternative access to the "Music News". This is done by associating 'Access' fragment "bcast://this.operator/access/952" with a 'Schedule' fragment without a 'defaultSchedule' attribute declared
        - the reference to "Music News" is instantiated as attribute 'idRef' under 'ContenReference' element to 'Content' fragment "bcast://this.operator/content/652".
        - the 'Content-Location' attribute under the 'ContenReference' element contains the 'Request-URI' used when "Music News" is to be streamed through the associated 'Access.
    - The 'Schedule' fragment with identifier "bcast://this.operator/schedule/553" declares that:
        - the contents "Music Video 1" and "Music Video 2" are available for the streaming delivery over the Interactive Channel.
        - the 'Access' fragment "bcast://this.operator/access/952", which describes the streaming delivery, is associated with 'Schedule' fragment through the reference within the 'Access' fragment.
        - This is the favourable schedule of the content item with identifier "bcast://this.operator/content/653" and "bcast://this.operator/content/654". This is done by setting the attribute 'defaultSchedule' in the 'Schedule' fragment to 'true'.
        - the references to two associated pieces of content are instantiated as individual 'ContenReference' elements with attribute 'idRef' to 'Content' fragments "bcast://this.operator/content/653" and "bcast://this.operator/content/654.
        - the 'Content-Location' attribute under each 'ContentReference' contains the 'Request-URI' used when "Music Video 1" and "Music Video 2" are to be streamed through the associated 'Access'
        - it is part of the "on demand" service and the associated content is not accessed automatically through this 'Schedule' fragment by the terminal when this 'Schedule' fragment is valid. This is done by setting the 'onDemand' attribute to 'true'.
        - it is not accessible through the main Schedule fragment with identifier "bcast://this.operator/schedule/551". This is also done by setting the 'onDemand' attribute to 'true'.
    - The 'Schedule' fragment with identifier "bcast://this.operator/schedule/554" declares that:
        - the contents "Music Video 2" is available for the download delivery over the Interactive Channel.
        - the 'Access' fragment "bcast://this.operator/access/953", which describes the download delivery, is associated with 'Schedule' fragment through the reference within the 'Access' fragment.
        - the reference to associated content is instantiated as attribute 'idRef' under 'ContenReference' elements to 'Content' fragments "bcast://this.operator/content/654".
        - the 'Content-Location' attribute under 'ContenReference' contains the 'Request-URI' used when "Music Video 2" is to be downloaded through the associated 'Access'.

- it is part of the "on demand" service and the associated content is not accessed automatically through this 'Schedule' fragment by the terminal when this 'Schedule' fragment is valid. This is done by setting the 'onDemand' attribute to 'true'.
- it is not accessible through the main Schedule fragment with identifier "bcast://this.operator/schedule/551". This is also done by setting the 'onDemand' attribute to 'true'.

  - The 'Schedule' fragment with identifier "bcast://this.operator/schedule/555" declares that:
    - the contents "Music Video 3" is available for the download delivery over the Interactive Channel.
    - the 'Access' fragment "bcast://this.operator/access/953", which describes the download delivery, is associated with 'Schedule' fragment through the reference within the 'Access' fragment.
    - the reference to associated content is instantiated as attribute 'idRef' under 'ContenReference' elements to 'Content' fragments "bcast://this.operator/content/655".
    - the 'Content-Location' attribute under 'ContenReference' contains the 'Request-URI' used when "Music Video 3" is to be downloaded through the associated 'Access'.
    - it is part of the "on demand" service and the associated content is not accessed automatically through this 'Schedule' fragment by the terminal when this 'Schedule' fragment is valid. This is done by setting the 'onDemand' attribute to 'true'.
    - it is not accessible through the main Schedule fragment with identifier "bcast://this.operator/schedule/551". This is also done by setting the 'onDemand' attribute to true.
  - Each of the mentioned 'Schedule' fragments have a reference to 'Service' fragment "bcast://this.operator/service/450" associating these schedules with the service "Music Channel".

- 'Access' fragment with identifier "bcast://this.operator/access/952" describes the access over the Interactive Channel to the contents associated with 'Schedule' fragments with identifiers "bcast://this.operator/schedule/552" and "bcast://this.operator/schedule/553", i.e. the contents "Music News" , "Music Video 1" and "Music Video 2". Within the 'Access' fragment the E2-element 'UnicastServiceDelivery' is instantiated to declare this access is over the Interactive Channel and the attribute 'type' is set to "3 - RTSP" to indicate this is a streaming delivery for which the stream setup and negotiation is done using the RTSP protocol.
  - The 'AccessServerURL' of this fragment is assigned with the value of server with which the terminal will establish the RTSP session.
  - Once set up, the terminal will request the corresponding content through the RTSP method by identifying the requested stream with value of attribute 'contentLocation' of element 'ContentReference' of 'Schedule' fragment
- 'Access' fragment with identifier "bcast://this.operator/access/953" describes the access over the Interactive Channel to the contents associated with 'Schedule' fragments with identifiers "bcast://this.operator/schedule/554" and "bcast://this.operator/schedule/555", i.e. the contents "Music Video 2" and "Music Video 3". Within the 'Access' fragment the E2-element 'UnicastServiceDelivery' is instantiated to declare this access is over the Interactive Channel and the attribute 'type' is set to "0 - HTTP" to indicate this is a download delivery.
  - The 'AccessServerURL' of this fragment is assigned with the value of server with which the terminal will establish the HTTP session.
  - Once set up, the terminal will request the corresponding content through the HTTP GET method by assigning the 'Request-URI' of HTTP method with value of attribute 'idRef' under 'ContenReference' of 'Schedule' fragment "bcast://this.operator/schedule/554".
- 'PurchaseItem' fragment with identifier "bcast://this.operator/pitem/151" is refers to 'Schedule' fragment with identifier "bcast://this.operator/schedule/551". This way the contents of "Music Channel" delivered over Broadcast Channel are defined as a purchasable item.
- 'PurchaseItem' fragment with identifier "bcast://this.operator/pitem/152" is refers to 'Schedule' fragments with identifiers "bcast://this.operator/schedule/552", "bcast://this.operator/schedule/553", "bcast://this.operator/schedule/554" and "bcast://this.operator/schedule/555". This way the contents of "Music Channel" delivered over Interactive Channel are defined as another purchasable item.

- Figure 30 above depicts the necessary Service Guide fragments and their relations.

Appendix D. Example of SGDD and SGDU

The following diagram provides an illustrative example of the relationship between SGDD and SGDU, as well as the containers in which they reside – SG Announcement Channel and SG Delivery Channels, respectively. In this example, it is assumed that the grouping of SG fragments is by time in the delivery over hourly, daily and weekly SG delivery channels.

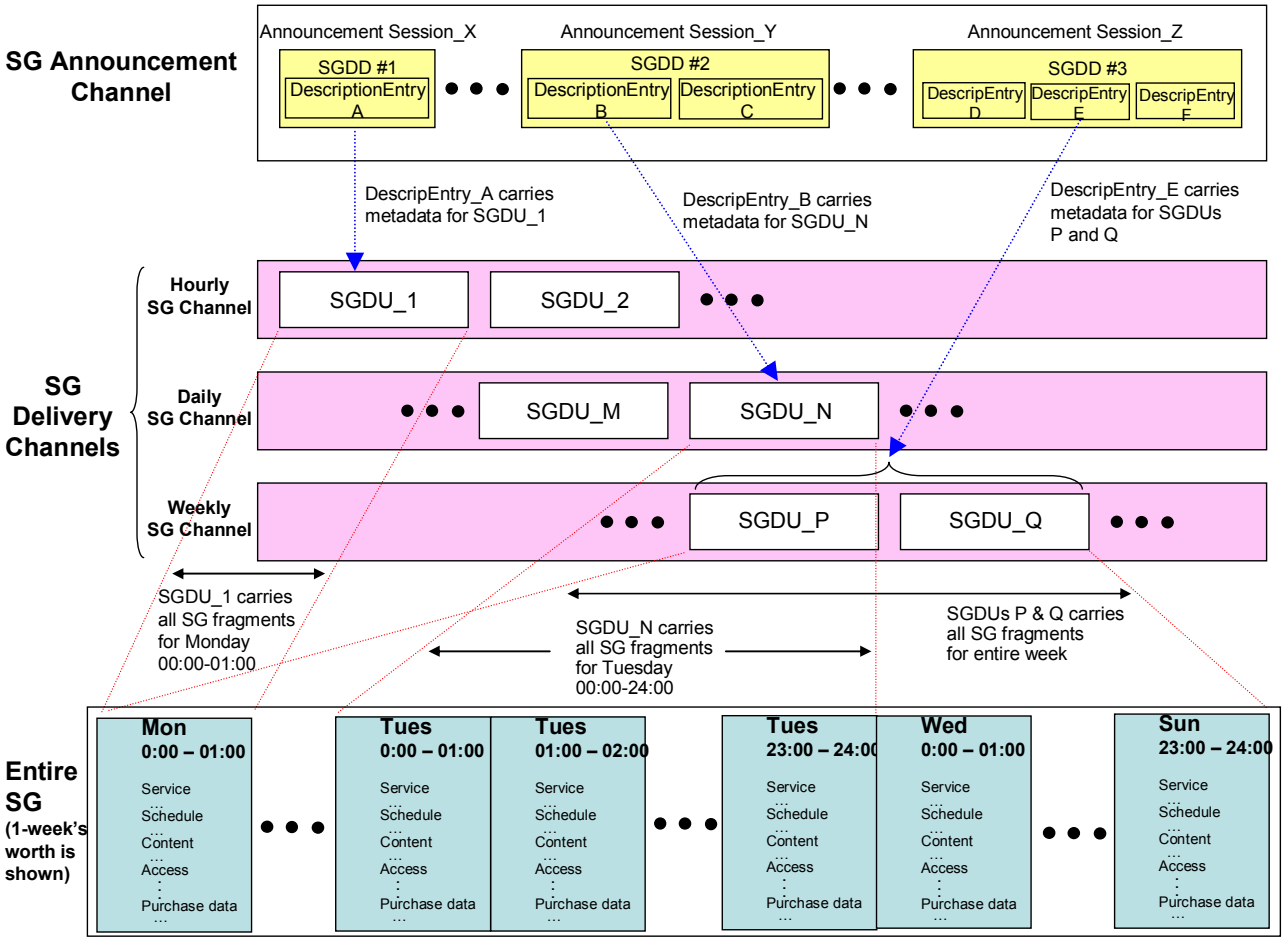


Figure 271: Service Guide Delivery Descriptor role in Service Guide delivery showing SG fragments grouped by time

Appendix E. Service Class Identifier Handling (Normative)

E.1 Definition of service class values

E.1.1 Service Class Definition

A service class identifier is a unique name used to identify a particular service class. The service class is a combination of many properties, such as used media formats, file types, handling of media, interaction with user, etc, which describes how a particular service works. A service class SHALL be documented in a specification, and a unique service class identifier SHALL be assigned to that service, following the syntax and procedure in F.1.2. Individual service providers may not have a formal specification for a service, however the particular capabilities of a specific implementation of the service is recommend to be provided with a service class identifier.

The service class identifier is provided with the 'Access' fragment and allows a terminal to match services and applications.

E.1.2 Service Class Naming and Registration

The Open Mobile Naming Authority [OMNA] maintains a registry of service classes. The service class names in this registry are structured as a URN to assure uniqueness. The model follows the following grammar:

{ClassName} ::= { *{OMAname}* | *{ExtName}* }

{OMAname} ::= **urn:oma:bcast:oma_bsc:***{ClassLabel}:{Version}*

{ExtName} ::= **urn:oma:bcast:ext_bsc_***{Org}:{ClassLabel}:{Version}*

Where:

- *{ClassLabel}* is a string made up of lowercase alpha characters
- *{Version}* is a string consisting of numbers and periods (e.g. '1.2')
- *{Org}* is a string that conveys the name of the external organization

Service classes defined in an OMA enabler SHALL be registered with [OMNA], providing a description of the service, a reference to the specification where it is documented and a service class identifier proposal following the above syntax for {OMAName}.

External organizations defining service classes are RECOMMENDED to register a URN prefix for their service class identifiers with [OMNA] and publish their service classes in their own registry. Uniqueness of service class identifiers can then be managed as the prefix will be unique to the registering organization and the {ClassLabel}:{Version} need only be unique within that organization's local registry.

To assist with communication of service classes defined by external organizations, [OMNA] will provide links to registration information available from these groups, as appropriate. Organizations seeking a prefix assignment are advised to check with [OMNA] for information.

E.2 Service Classes defined in OMA BCAST

This section defines the service classes for for the OMA BCAST enabler.

E.2.1 urn:oma:bcast:oma_bsc:sg:1.0

Services of this service class deliver metadata fragments using a broadcast channel or the interactive channel. The discovery of Service Guide is defined in section 6.

E.2.2 urn:oma:bcast:oma_bsc:fc:1.0

Services of this service class offer a file carousel service, using the File Delivery function of OMA BCAST. Files are transmitted and locally stored in a file cache. Files are uniquely identified by the URI. Files in the file carousel are periodically re-transmitted. It may happen that already received files are replaced by a new file in a later repetition. An example of a file carousel is a teletext service.

E.2.3 urn:oma:bcast:oma_bsc:tp:1.0

Services of this service class of offer a Terminal Provisioning service either using the broadcast channel over TP-5 or interaction channel over TP-7. The management objects delivered over sdo.oma.tp SHALL be of MIME type “application/vnd.syncml.dm+wbxml". [BCAST10-Services] defines Terminal Provisioning in section 5.2.

E.2.4 urn:oma:bcast:oma_bsc:nt:1.0

Services of this service class deliver Notification Messages over NT-5 or NT-6. Both over the broadcast channel as well as over the interactive channel, the delivered messages SHALL follow the format as specified in section 5.14 of [BCAST10-Services]. The discovery and signaling of Notification Messages SHALL follow the specification in section 5.14.1 of [BCAST10-Services].

E.2.5 urn:oma:bcast:oma_bsc:csg:1.0

Services of this service class deliver a complementary Service Guide. The discovery of Service Guide is defined in section 6.

E.2.6 urn:oma:bcast:oma_bsc:st:1.0

Services of this service class offer a streaming service, using the Stream Delivery function of OMA BCAST. Streams are transmitted as specified in section 6 of [BCAST10-Distribution]. Examples of this service are live TV and live Radio, realized as streaming audiovisual or audio-only services.

E.2.7 urn:oma:bcast:oma_bsc:rifc:1.0

Services of this class offer a file carousel service for transmission of Registration Layer and Rights Management Layer objects and messages. RI Services are specified in section 12 of [DRM20-Broadcast-Extensions].

Appendix F. Global Status Codes

The Global Status Codes specified in [BCAST10-Services] lists all the possible status codes for error case, and their applicability to each transaction. The codes are to be used for Response in SG-4 and NT-1, NT-3, NT-4 response messages in this specification. The codes may also be used in other response messages in other BCAST technical specifications.

The Table 8 below shows example values from the Global Status Codes specified in [BCAST10-Services] for the transaction messages that require the use of Global Status Codes. The values shown below are for informative purposes and the full range of values of the Global Status Codes are applicable to all messages if deemed required.

TS-BCAST_ServiceGuide	5.6.2.2.2 Response Message for content delivery	000, 007, 008, 011, 012, 015, 016, 017, 018, 019, 020, 021, 022, 023
	5.6.2.3 Response Message for service protection information delivery	000, 007, 008, 011, 012, 015, 016, 017, 018, 019, 020, 021, 022, 023
	5.6.2.4 Response Message for service provisioning information delivery	000, 007, 008, 011, 012, 015, 016, 017, 018, 019, 020, 021, 022, 023
	7.5. 2.2 Response Message	000, 007, 008, 011, 012, 015, 016, 017, 018, 019, 020, 021, 022, 023
	7.5.3.2 Response Message	000, 007, 008, 011, 012, 015, 016, 017, 018, 019, 020, 021, 022, 023

Table 8: Cross Reference Table (Informative)

Appendix G. MIME media types

G.1 Media-Type Registration Request for application/vnd.oma.bcast.sgdu

This section provides the registration request, as per [RFC 2048], to be submitted to IANA.

Type name: application

Subtype name: vnd.oma.bcast.sgdu

Required parameters: none

Optional parameters: none

Encoding considerations: binary

Security considerations:

BCAST Service Guide Delivery Unit data shall be passive, and does not generally represent a unique or new security threat. However, there is some risk in sharing any kind of data, in that unintentional information may be exposed, and that risk applies to data contained in Service Guide Delivery Unit as well.

Interoperability considerations:

This content type carries service guide fragments within the scope of the OMA BCAST enabler. The OMA BCAST enabler specification includes static conformance requirements and interoperability test cases for this content.

Published specification:

OMA BCAST 1.0 Enabler Specification – Service Guide for Mobile Broadcast Services, especially section 5.4.1.3. Available from http://www.openmobilealliance.org

Applications, which use this media type:

OMA BCAST Service Guide

Additional information:

Magic number(s): none

File extension(s): none

Macintosh File Type Code(s): none

Intended usage: Limited use.

Only for usage with BCAST Service Guide Delivery Unit, which meet the semantics given in the mentioned specification.

Person & email address to contact for further information:

Uwe Rauschenbach

Uwe.Rauschenbach@nsn.com

Author/Change controller:

OMNA – Open Mobile Naming Authority, OMA-OMNA@mail.openmobilealliance.org

G.2 Media-Type Registration Request for application/vnd.oma.bcast.sgdd+xml

This section provides the registration request, as per [RFC 2048], to be submitted to IANA.

Type name:	application
Subtype name:	vnd.oma.bcast.sgdd
Required parameters:	none
Optional parameters:	none
Encoding considerations:	binary

Security considerations:

BCAST Service Guide Delivery Descriptor data shall be passive, and does not generally represent a unique or new security threat. However, there is some risk in sharing any kind of data, in that unintentional information may be exposed, and that risk applies to data contained in Service Guide Delivery Descriptor as well.

Interoperability considerations:

This content type carries service guide delivery information within the scope of the OMA BCAST enabler. The OMA BCAST enabler specification includes static conformance requirements and interoperability test cases for this content.

Published specification:

OMA BCAST 1.0 Enabler Specification – Service Guide for Mobile Broadcast Services, especially section 5.4.1.5. Available from http://www.openmobilealliance.org

Applications, which use this media type:

OMA BCAST Service Guide

Additional information:

Magic number(s):	none
File extension(s):	none

Macintosh File Type Code(s): none

Person & email address to contact for further information:

Uwe Rauschenbach

Uwe.Rauschenbach@nsn.com

Intended usage: Limited use.

Only for usage with BCAST Service Guide Delivery Descriptor, which meet the semantics given in the mentioned specification.

Author/Change controller: OMNA – Open Mobile Naming Authority, OMA-OMNA@mail.openmobilealliance.org

Appendix H. Appendix H.Identification of unique elements (Normative)

H.1 Definition of BCAST Identification Authority

In order to permit unique identification of items in the Service Guide, items URI have to be constructed in a way that allows such unique identification. In the context of BCAST, such responsibility is delegated to Identification Authorities who are responsible for providing unique identifiers within their domain of control. Global uniqueness of items is then ensured by having the Identification Authority registered in a common registry.

The Open Mobile Naming Authority [OMNA] maintains such registry of Identification Authorities. The Identification Authority name in this registry is structured as a URN to ensure uniqueness. The model follows the following grammar:

{IdentAuthLabel} ::= urn:oma:bcast:iauth:{*IAiD*}

Where *{IAiD}* is a string assigned to a BCAST Identification Authority. It is made up of lowercase alphanumeric characters and the characters "-" and "." where the latter two characters are not allowed at the first and last positions in the string.

Organizations willing to act as Identification Authorities providing item identification in the BCAST Service Guide are advised to register with [OMNA]. To assist in this, [OMNA] will provide links to registration information available as appropriate. Organizations seeking a prefix assignment are advised to check with [OMNA] for information. [OMNA] registered *{IAiD}* values are restricted to lowercase alphanumeric characters.

Note 1: for testing purposes, the pattern "x-*", where "*" is a string of lowercase alphanumeric characters, is reserved for the *{IAiD}* in the OMNA registry. This pattern is not open for registration and can be used for private values.

Note 2: an alternative to registering an *{IAiD}* with [OMNA] is to use a domain name registered to the Identification Authority as part of the *{IAiD}*. In such case, the format of *{IAiD}* SHALL be "dns-*" where "*" is a valid domain name according to [RFC 1035] section 2.3.1, with the update defined in [RFC 1123] section 2.1. It should be noted, however, that using domain names can cause collision problems for items with a long lifetime when the owner of the domain name changes. This pattern is not open for registration.

H.2 Identification of content with baseCID

Unique identification of one content is done by an Identification Authority, extending its own identification URN into a URI. The model follows the following grammar:

{BaseCID} ::= *{IdentAuthLabel}*:**basecid**:*{BaseCIDExt}*

Where :

- *{IdentAuthLabel}* is the URN of the Identification Authority as defined in section H.1.
- *{BaseCIDExt}* is an extension string that identifies the content within the scope of the Identification Authority.
- *{BaseCID}* provides a fully qualified identification of the content that is usable as a 'baseCID' in the 'Service' and 'Content' fragment.