# BCAST Distribution System Adaptation – 3GPP2/BCMCS

## Candidate Version 1.2 – 14 Jan 2014

**Open Mobile Alliance**

OMA-TS-BCAST_BCMCS_Adaptation-V1_2-20140114-C

Use of this document is subject to all of the terms and conditions of the Use Agreement located at http://www.openmobilealliance.org/UseAgreement.html.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. Information contained in this document may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification. However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the "OMA IPR Declarations" list at http://www.openmobilealliance.org/ipr.html. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR'S REPRESENTED ON THE "OMA IPR DECLARATIONS" LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

# Contents

# Figures

# Tables

# 1. Scope

This document specifies how the BCAST 1.2 enabler is implemented over a specific BDS (BCAST Distribution System).

The BCAST 1.2 Enabler supports the global interoperability among different BCAST Distribution Systems, and can also be adapted according to the characteristics of BCAST Distribution Systems for BCAST 1.2 enabler implementation over a certain BDS.  In this document, two types of adaptation are presented.

The BCAST 1.2 Enabler includes 9 functions and all 9 functions BCAST functions can be used implemented over the specific BDS without any use of underlying functionality with minimal adaptation is needed to achieve this. This is referred to as "generic adaptation", which can be applied for any kind of BDS.

The underlying BDS may already have a method for a function defined in the BCAST 1.2 Enabler. This specification also enables this native BDS method to be utilised for the BCAST function so as to enable interoperability with existing services. In this case the BCASTadaptation mode is referred to as "BDS specific adaptation".

This is further explained in Section 4 - Introduction.

# 2. References

## 2.1 Normative References

| | |
|---|---|
| **[3GPP2 C.R1001-G]** | "Administration of Parameter Value Assignments for cdma2000 Spread Spectrum Standards", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 C.R1001-G, Release G, Version 1.0, June, 2009, URL: http://www.3gpp2.org |
| **[3GPP2 C.S0023-D]** | "Removable User Identity Module for Spread Spectrum Systems", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 C.S0023-D, Release D, Version 1.0, June 2009, URL: http://www.3gpp2.org |
| **[3GPP2 C.S0024-B]** | "cdma2000 High Rate Packet Data Air Interface Specification", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 C.S0024-B, Release B, Version 3.0, September 2009, URL: http://www.3gpp2.org |
| **[3GPP2 C.S0054-A]** | "cdma2000 High Rate Broadcast-Multicast Packet Data Air Interface Specification", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 C.S0054-A, Release A, Version 1.0, February 2006, URL: http://www.3gpp2.org |
| **[3GPP2 C.S0070-0]** | "Broadcast Multicast Services (BCMCS) Codecs and Transport Protocols", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 C.S0070-0, Release 0, Version 1.0, Spring 2010, URL: http://www.3gpp2.org |
| **[3GPP2 S.S0083-A]** | "Broadcast-Multicast Service Security Framework", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 S.S0083-A, Release A, Version 1.0, August 2004, URL: http://www.3gpp2.org |
| **[3GPP2 X.S0002-0]** | "TIA/EIA-41-D Location Services Enhancements", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 X.S0002-0, Release 0, Version 1.0, March 2004, URL: http://www.3gpp2.org |
| **[3GPP2 X.S0022-A]** | "Broadcast and Multicast Service in cdma2000 Wireless IP Network", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 X.S0022-A, Release A, Version 1.0, February 2007, URL: http://www.3gpp2.org |
| **[3GPP2 X.S0024-0]** | "IP-Based Location Services, Revision 0", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 X.S0024-0, Release 0, Version 1.0, October 2005, URL: http://www.3gpp2.org |
| **[3GPP2-S.R0030-A]** | "Broadcast/Multicast Services-Stage 1, Revision A", 3rd Generation Partnership Project 2, Technical Specification 3GPP2 C.R0030-A, Release A, Version 1.0, January 2004, URL: http://www.3gpp2.org |
| **[BCAST11-Architecture]** | "Mobile Broadcast Service Architecture," Version 1.1, Open Mobile Alliance™, OMA-AD-BCAST-V1_1, URL: http://www.openmobilealliance.org |
| **[BCAST12-Distribution]** | "File and Stream Distribution for Mobile Broadcast Services", Version 1.2, Open Mobile Alliance™, OMA-TS-BCAST_Distribution-V1_2, URL: http://www.openmobilealliance.org/ |
| **[BCAST12-ServContProt]** | "Service and Content Protection for Mobile Broadcast Services," Version 1.2, Open Mobile Alliance™, OMA-TS-BCAST_SrvCntProtection-V1_2, URL: http://www.openmobilealliance.org |
| **[BCAST12-Services]** | "Mobile Broadcast Services", Version 1.2, Open Mobile Alliance™, OMA-TS-BCAST_Services-V1_2, URL: http://www.openmobilealliance.org/ |
| **[BCAST12-SG]** | "Service Guide for Mobile Broadcast Services," Version 1.2, Open Mobile Alliance™, OMA-TS-BCAST_Service_Guide-V1_2, URL: http://www.openmobilealliance.org |
| **[DRM20-XBS]** | "OMA DRM v2.0 Extensions for Broadcast Support", Open Mobile Alliance™, OMA-TS-DRM-XBS-V1_2, URL: http://www.openmobilealliance.org/ |

| **[IOPPROC]** | "OMA Interoperability Policy and Process", Version 1.1, Open Mobile Alliance™, OMA-IOP-Process-V1_1,<br>URL: http://www.openmobilealliance.org |
|---|---|
| **[OMA MLP]** | "Mobile Location Protocol," Version 3.2, Open Mobile Alliance™,<br>URL: http://www.openmobilealliance.org |
| **[RFC 2119]** | "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997,<br>URL: http://www.ietf.org/rfc/rfc2119.txt |
| **[RFC 2234]** | "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997,<br>URL: http://www.ietf.org/rfc/rfc2234.txt |
| **[RFC 4234]** | "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. October 2005,<br>URL: http://www.ietf.org/rfc/rfc4234.txt |
| **[RFC 4566]** | "SDP: Session Description Protocol", M. Handley, V. Jacobson and C. Perkins, July 2006,<br>URL: http://www.ietf.org/rfc/rfc4566.txt |
| **[SCRRULES]** | "SCR Rules and Procedures", Open Mobile Alliance™, OMA-ORG-SCR_Rules_and_Procedures,<br>URL:http://www.openmobilealliance.org/ |

## 2.2    Informative References

| **[OMADICT]** | "Dictionary for OMA Specifications", Version x.y, Open Mobile Alliance™,<br>OMA-ORG-Dictionary-Vx_y, URL:http://www.openmobilealliance.org/ |
|---|---|

# 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.

This is an informative document, which is not intended to provide testable requirements to implementations.

## 3.2 Definitions

| | |
|---|---|
| BCAST Distribution System | A system typically but not necessarily containing the ability to transmit the same IP flow to multiple Terminal devices simultaneously. A BCAST Distribution System (BDS) typically uses techniques that achieve efficient use of radio resources. A BDS consists of Network functionality up to the IP layer and optional Service Distribution/Adaptation functionality above the IP layer. Most BDSs support broadcast/multicast distribution in the network. Some BCAST Distribution Systems have the capability to deliver the IP flows in the network via unicast. |
| BCMCS Content Server | A BCMCS entity that is a function that makes BCMCS content available within an IP Multicast stream. The BCMCS Content Server in the serving network is not necessarily the creator or source of the content. It is the last application level entity to manipulate (e.g., reformat) the content prior to the content reaching the BSN. The BCMCS Content Server may store and forward the content from the BCMCS Content Provider, and/or merge the content from the multiple content providers. If higher layer encryption is enabled, the BCMCS Content Server encrypts the stream content. In this case, the BCMCS Content Server also serves the function of SK manager. [3GPP2 X.S0022] |
| BCMCS Controller | A BCMCS entity that is a core network function that is responsible for managing and providing the BCMCS session information to the BSN (via the SAAA), the BCMCS Content Server, the MS (via the PDSN function) and the RAN (via the BSN function). The BCMCS Controller serves the function of BAK Distributor and may serve the function of BAK Generator. The BCMCS Controller may perform discovery operations to assist the MS to find desired content. [3GPP2 X.S0022] |
| BCMCS Program | A BCMCS Program can represent either a native BCMCS service or a BCAST enabled service, or a mixture of both. |
| BCMCS Services | A broadcast or multicast service that is provided by BCMCS |
| BCMCS Subscriber Profile Manager | A BCMCS entity that is an application that updates the subscriber profile in the Databases regarding subscribed BCMCS services. The user may interface to this application directly, or the operator may reserve access to this application to their customer service agents. The interface between the user and the BCMCS Subscriber Profile Manager, and the interface between the BCMCS Subscriber Profile Manager and the Subscriber Profile Database are outside the scope of this Document. [3GPP2 X.S0022] |
| Smartcard | A non-UICC secure function platform which may contain the SIM or R-UIM module, or a UICC-based secure function platform which may contain one or more of the following applications: a 3GPP USIM, 3GPP2 CSIM or 3GPP/3GPP2 ISIM. Note that the set of applications/modules residing on the Smartcard are typically governed by the affiliation of the Smartcard to 3GPP or 3GPP2 specifications, as indicated by the definition below for "Smartcard Profile". |

## 3.3 Abbreviations

| | |
|---|---|
| **(R-)UIM/CSIM** | (Removable -) Universal Identity Module/cdma2000® SIM |
| **BCMCS** | 3GPP2 Broadcast Multicast Service |
| **BDS** | BCAST Distribution System |
| **BSD/A** | BCAST Service Distribution and Adaptation |
| **Cell_Group_ID** | Service Area Identifier or Location Area as described in [3GPP2 X.S0002] |
| **Cell_ID** | Cell identifier as defined in [3GPP2 X.S0002] |
| **CS** | Content Server |

| **MKI** | Master Key Index |
|---------|------------------|
| **MLP** | Mobile Location Protocol |
| **OMA** | Open Mobile Alliance |
| **QoS** | Quality of Service |
| **RK** | Registration Key |
| **ROC** | Rollover Counter |
| **SG** | Service Guide |
| **SGDD** | Service Guide Delivery Descriptor |
| **SGDU** | Service Guide Delivery Unit |
| **SRTP** | Secure Real Time Protocol |
| **TE** | Terminal Equipment |

# 4. Introduction

This document technical specification specifies how the OMA Mobile Broadcast Services (BCAST) Enabler can be implemented in 3GPP2 BCMCS network.

## 4.1    Version 1.0

BCAST ERP 1.0 has two modes of adaptation for BCMCS:

1.    Generic adaptation over an underlying BCMCS transport network

In this situation mode, this Technical Specification the adaptation specification explains how the BCAST Enabler has access to the IP transport layer so that BCAST services can be provided transparently from BCAST Network entities server to terminal BCAST Terminal. In generic adaptation mode, BCAST Network entities and BCAST Terminal conform to BCAST main specifications. This means the BCAST terminal and server conform to the main BCAST technical specifications. Furthermore, this allows a common, harmonised behaviour across multiple BCAST enabled BCAST Distribution Systems (BDSes) over which BCAST services are deployed.

However, in Generic adaptation mode, it may be impossible to share in most cases this means corresponding BCMCS functionality is not re-used and in most cases broadcast services can not be shared with a native BCMCS server / terminal due to differences between the technologies selected in the specific BDS and the because some of BDS specific technologies may not be implemented by Generic adaptation. For example, file delivery mechanisms may be different or service and content protection mechanisms may be different. In practice this means file delivery sessions and streaming sessions are most likely to be simulcasted provided in parallel in order to cater for BCAST Terminals and native BDS BCMCS terminals.

2.    BDS specific adaptation to BCMCS functionality

In this situation mode,  this Technical Specification explains how the adaptation specification provides adaptation of various BCAST functionalities are adapted in a BCMCS network taking in consideration the specific technical aspects of the underlying BCAST Distribution System (BDS). In this mode, it is possible that this means that BCAST server and terminal re-use underlying functionality so that broadcast services can be shared between BCAST terminals and native BCMCS terminals. Hence BCAST Network entities and BCMCS servers can provide services to both types of terminals.

For example, file delivery mechanisms and protection mechanisms would be those defined by native 3GPP2 BCMCS specifications. In practice this means file delivery sessions and streaming sessions  would cater for both BCAST terminals and BCMCS terminals, without the need for providing sessions in parallel simulcasting (unlike 1 above).

A consequence of adaptation to the underlying BDS functionality is that the BCAST behaviour is profiled to each BDS, making it difficult or impossible to share BCAST services across multiple BDS's.

This document specifies how both types of behaviour can be achieved.

Note that the purpose of BDS specific adaptation is to enable sharing a service between BCAST terminals and native BDS terminals. In contrast, generic adaptation allows sharing a BCAST service across different BDSs. As described above, Hence BCAST Servers Network entities and BCAST Terminals will be able to handle the two types of adaptation, providing maximum deployment flexibility for the Service Provider. This allows BCAST terminal to work automatically in both situations, as signalling is provided to indicate to the terminal the type of adaptation provided. As not all underlying BDS functionality is re-used by BCAST, BCAST services Enabler may be adapted to both types of adaptation, i.e. BDS specific adaptation (re-using underlying BDS functionality optimized for BDS) for certain functions whilst using generic adaptation (BCAST-specific functionality) for other functions.

## 4.2    Version 1.1

There are no changes to Version 1.1, compared to Version 1.0.  Only the references, hyperlinks, and grammar in this document have been updated.

## 4.3    Version 1.2

There are no changes to version 1.2, compared to version 1.1.  Only the references in this document have been updated.

# 5. Overview of 3GPP2 BCMCS

BCMCS provides point-to-multipoint transmission of multimedia data (e.g., text, audio, pictures, video) from a single source to all users or a group of users in a specific area. The BCMCS system design aims to satisfy the market demand for broadcast and multicast content while minimizing resource usage in the radio access network (RAN).

The cellular service provider or content provider indicates the availability of BCMCS to users via BCMCS service announcement and discovery. This mechanism enables the network to inform users about services available. Service discovery mechanisms allow users to request information about available BCMCS services from the network. Mobile users who desire BCMCS service may discover the BCMCS content and schedule via various mechanisms such as advertisements, short messaging service (SMS), HTTP-based web access, etc. The BCMCS Controller may act as a server to provide the mobile station with information on BCMCS content and schedules. Service discovery/announcement is used to distribute to users information about the services (e.g., content name or multicast IP addresses and port numbers for particular content programs) and possibly other service-related parameters (e.g., service registration allowed time, service start and end times).

Upon discovering the services, a mobile user who wishes to receive certain BCMCS programs must subscribe with the service provider. As part of the subscription process, a shared secret, known as the Registration Key (RK) is provisioned in the user identification module (i.e. (R-)UIM or CSIM) and the service provider's subscription database. Upon subscription, the terminal performs BCMCS information acquisition procedures to acquire necessary information on the BCMCS session, header compression, and transport and application protocols to be able to receive BCMCS programs. One BCMCS program may consist of multiple multicast IP flows, for example, audio and video streams.

After BCMCS information acquisition, the terminal determines whether a desired multicast IP flow is available in a particular cell and sector by obtaining the corresponding radio configuration information from a base station via overhead messages on the control channel. If the BCMCS bearer path is not yet established, the first terminal performing BCMCS registration may trigger the PDSN (Packet Data Serving Node) to join the multicast group associated with the BCMCS_FLOW_IDs, to subsequently set up a bearer path from the RAN to the PDSN. This mode of operation makes more efficient use of air interface resources; by eliminating the need for multiple terminals to each send multicast join messages over the air.

When the network determines that there are no more terminals listening to a specific multicast IP flow(s), it may release the associated bearer path. The network may also release the bearer resources when the scheduled BCMCS program is finished.

The set of specification documents applicable to BCMCS consists of the following 3GPP2 specifications:

- General description and functional requirements are provided in [3GPP2 S.R0030-A].

- BCMCS security framework is specified by [3GPP2 S.S0083-A]

- Network architecture, BCMCS service description, terminal-to-network and network-to-network interactions, security, usage accounting, and protocols and message formats are specified by [3GPP2 X.S0022-A].

- cdma2000 1X air interface physical, MAC and signalling layers are specified by [3GPP2 C.S0001-D through C.S0005-D].

- cdma2000 1X air interface related BCMCS service aspects, and segment-based framing are described by [3GPP2 C.S0077-0].

- cdma2000 HRPD BCMCS air interface is specified in [3GPP2 C.S0024-B].

- cdma2000 HRPD BCMCS related protocols, service aspects, and segment-based framing is specified by [3GPP2 C.S0054-A].

- cdma2000 Broadcast Multicast Services (BCMCS) Codecs and Transport Protocols specifications are specified by [3GPP2 C.S0070-0].

- cdma2000 IP-Based Location Services are specified by [3GPP2 X.S0024-0].

- Removable User Identity Module for Spread Spectrum Systems is specified by [3GPP2 C.S0023-D].

- Administration of Parameter Value Assignments for cdma2000 Spread Spectrum Standards as specified by [3GPP2 C.R1001-G].

- TIA/EIA-41-D Location Services Enhancements as specified by [3GPP2 X.S0002-0].

# 6. Generic Adaptation over BCMCS' IP Transmission Network

This section describes how BCAST specifications (namely, [BCAST12-Services], [BCAST12-SG], [BCAST12-ServContProt], [BCAST12-Distribution] and [DRM20-XBS]) are used over a BCMCS network.  The provisions in this section thus complement the ones in the generic specifications so that BCAST services can be distributed over the BCMCS IP transmission network, without re-using the BCMCS functionality and hence without the ability for sharing services with native BCMCS terminals (unlike the BDS specific adaptation specified in Section 7).

All normative statements in this specification are only applicable in the case OMA BCAST services are distributed over BCMCS's IP transmission network specified in [3GPP X.S0022].

The sentence "as defined by BCAST Enabler specifications" is a shorthand notation that indicates both BCAST server and terminal SHALL respect the relevant BCAST specification (listed above).

Generic adaptation MAY be supported by BCAST Network entities and SHALL be supported by BCAST Terminal.

The message flows and architecture diagrams as shown in this section presume logically separate BCAST and BCMCS network-side functions.  In deployment, BCAST and BCMCS network functions may be integrated into one or more physical entities, in which case some of the protocol interfaces and associated message flows are not applicable (i.e., those corresponding reference points are internal to the  network entity(entities) and don't correspond to physical interfaces).

Signaling of the generic adaptation mode is performed using service provisioning procedures as defined in [BCAST12-Services].

## 6.1    IP transport of BCAST

This functionality is specified in [3GPP2 X.S0022-A].

## 6.2    Generic adaptation related to OMA-TS-BCAST_Services

### 6.2.1    Interaction

Since the interaction channel exists and is used in BCMCS, the BCAST Terminal SHALL support all four types of interaction defined by the generic BCAST Enabler specifications (namely, [BCAST12-Services], [BCAST12-SG] and [BCAST12-Distribution]).

### 6.2.2    Service Provisioning

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-Services], Section 5.1).

### 6.2.3    Terminal Provisioning

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-Services], Section 5.2 and Appendix E).

### 6.2.4    Notification

The specification in Section 5.14 of [BCAST12-Services] SHALL apply.

When using 3GPP2 BCMCS as the underlying BCAST Distribution System the Notification functionality is enabled as specified in [BCAST12-Services].

## 6.3    Generic adaptation related to OMA-TS-BCAST_ServiceGuide

### 6.3.1    Service Guide Delivery over Broadcast Channel

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.4.2).

### 6.3.2    Service Guide Delivery over Interaction Channel

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.4.3).

## 6.3.3    Service Guide Encoding

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.4.1).

## 6.3.4    Session Description Information

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.1.2.5), with the following restrictions:

– The session information SHALL be provided using an SDP-formatted file contained in the Access fragment or in a Session Description referenced by the Access fragment.

– The MBMS User Service Bundle Description/User Service Description SHALL NOT be used.

A BCAST Terminal supporting BCMCS SHALL support SDP [RFC 4566].

### 6.3.4.1    Session Description parameters for broadcast streamed media sessions

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.1.2.5.2).  The terminal MAY ignore the following parameters in Section 5.1.2.5.2 of [BCAST12-SG] should these be present in the SG, as they are either not required or out of the scope of BCMCS:

- The mode of MBMS bearer per media

- Initial buffering delay, using the min-buffer-time attribute as specified in [DVB JTC 188] section 5.3.4.

### 6.3.4.2    Session Description parameters for broadcast file delivery sessions

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.1.2.5.3) but with qualifications as indicated below.

#### 6.3.4.2.1    Session Descriptors for ALC Sessions

The terminal MAY ignore the following parameter, should it be present in the SG, as it is outside the scope of BCMCS:

- The mode of MBMS bearer per media

#### 6.3.4.2.2    Session Descriptors for FLUTE Sessions

The terminal MAY ignore the following parameter, should it be present in the SG, as it is outside the scope of BCMCS:

- The mode of MBMS bearer per media

## 6.3.5    Service Guide Data Model

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.1).

## 6.3.6    CellTargetArea/BDSLocationID in BCMCS

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Sections 5.1.2.1 and 5.1.2.3). BDSLocationID as specified in [BCAST12-SG, Section 5.4.3.4] reuses the structure of  the "value" attribute of  CellArea as defined in [BCAST12-SG].

## 6.3.7    BDSSpecificEntryPointInfo definition

Section 5.4.1.5.2 of [BCAST12-SG] specifies how SGDDs can include the definition of SGEntryPoints over BCAST BDS broadcast channels. Each broadcast SGEntryPoint (i.e. SG Announcement Channel) in a BCAST BDS is declared partially by generic parameters (such as 'srcIpAddress', 'port', etc.) and partially by BDS-specific parameters, provided in each BDS Adaptation TS via the extension by derivation of the abstract type of BDSSpecificEntryPointInfo element.  Note that the BDSSpecificEntryPointInfo is typically a subset of the BDSEntryPoint of the OMA BCAST Device Management (DM) Tree defined in the appendix of [BCAST12-Services].

For the BCMCS BDS, the abstract type of BDSSpecificEntryPointInfo element is derived as follows:

| Name | Type | Category | Cardinality | Description | Data Type |
|------|------|----------|-------------|-------------|-----------|
| BDSSpecific EntryPointInf o | E5 | NM/TM | 0..1 | The placeholder for the supplementary information that is required in order to retrieve the broadcast SG entry point in BCAST BDS, i.e. in BCMCS BDS for the present specification. | complexType deriving from abstract type of |

| | | | | | | BDSSpecific EntryPointInfo element |
|---|---|---|---|---|---|---|
| | | | | | Contains the following elements NetworkID ControllerIPAddress ROHCU | |
| NetworkID | E6 | NM/TM | 1 | | Network region identification information for BCMCS. The value of this node is a string comprising the concatenation of 1 (in the case of subnet) or 1, 2, or 3 (in the case of SID/NID/PZID) hexdecimal numbers, each number prefixed by characters 's' (subnet), 'S' (SID), 'N' (NID), or 'P' (PZID), used by the terminal to determine when overhead channels indicate that the terminal is in a particular network region. | string |
| ControllerIP Address | E6 | NM/TM | 1 | | The IP address of the BCMCS Controller, saving DHCP bootstrap time. | string |
| ROHCU | E6 | NM/TM | 0..1 | | This element carries parameters for the ROHC Unidirectional Protocol.  Contains the following attributes largeCIDs  Contains the following elements MaxCID MaxHeaderSize MRRU | |
| largeCIDs | A | NM/TM | 0..1 | | True when large CIDs (1 or 2 bytes) are used, otherwise it is false and small CIDs (0 or 1 bytes) are used. If this is absent it is assumed to be false. | boolean |
| MaxCID | E7 | NM/TM | 1 | | The maximum number of CIDs used by ROHC-U | unsignedint |
| MaxHeader Size | E7 | NM/TM | 1 | | The maximum header size, in octets, that can be compressed | unsignedInt |
| MRRU | E7 | NM/TM | 1 | | The size of the Maximum Reconstructed Reception Unit, in octets, that the decompressor is expected to reassemble from segments. Value 0 means that no segment headers are allowed on the channel | unsignedInt |

**Table 1: BSDSpecificEntryPointInfo Definition**

## 6.3.8   Service Guide Discovery

### 6.3.8.1     Service Discovery assisted by BCMCS information acquisition

The native BCMCS service discovery mechanism described below SHALL be employed in performing the BCAST Service Guide bootstrapping and service discovery procedures.

BCMCS defines an Information Acquisition mechanism which can be used to perform service discovery, similar to the acquisition of the BCAST Service Guide as defined in [BCAST12-SG].  BCMCS Information Acquisition enables the discovery of BCMCS application and transport-related information such as program name, schedule, description, IP address and port numbers of the BCMCS flows which carry the corresponding BCMCS program.  Therefore, as shown below, BCMCS Information Acquisition is used for service discovery of BCAST-enabled broadcast services based on the core concept that the BCAST Service Guide Delivery Descriptor (SGDD) and Service Guide (SG) are presented as separate BCMCS programs.

In the following call flow (Figure 1), it is assumed that BCMCS subscription, which provisions the RK in the terminal, and BCMCS Controller discovery, have already taken place as described in [3GPP2 X.S0022-A].

TE | BCMCS Content Server | BCMCS Controller | BCMCS Subscriber Profile Manager | OMA BCAST BSD/A

1. BCMCS Content Subscription (Program: SGDD, SG)

2. BCMCS Flow Provisioning (BCAST SGDD, SG, Services)

3. BCMCS Flow Provisioning

4. BCMCS Info Acquisition

5a. Pre-configure the entry point (IP/Port) of SG Announcement channel that carries SGDD

5b. Static Mapping between BCMCS Flow ID and IP/Port

6. SGDD Delivery

7. Static Mapping between BCMCS Flow ID and IP/Port for SGDU and SG

8. SG Delivery

**Figure 1: BCMCS Service Discovery of BCAST SGDD and SG as BCMCS Programs**

1. The user subscribes to the BCMCS programs for SGDD, SG with the BCMCS Subscriber Profile Manager via out of band mechanisms.

2. The OMA BCAST BSD/A performs flow provisioning with BCMCS Controller for BCAST SGDD and SG as separate BCMCS programs. In flow provisioning, the IP Address/ port, schedule, QoS for BCMCS programs are provided.

3. The BCMCS Controller performs flow provisioning with the BCMCS CS (see [3GPP2 X.S0022-A]).

4. Terminal performs BCMCS Information Acquisition with the BCMCS controller to discover BCMCS application and transport related information for SGDD such as program name and description and IP address and port number of the SG Announcement Channel which delivers SGDDs. BCMCS Information Acquisition also provides the mapping between IP address/port number to BCMCS Flow ID. Note that Step 4 is unnecessary should the terminal be pre-configured with the IP address and port number of the SG Announcement Channel, and can perform static mapping between IP/port and Flow ID.

5. a) and b) presumes the terminal is pre-configured with the IP address and port number of the SG announcement channel which transmits SGDD(s), and that the IP address and port are within the range of static mapping (see X.P0022) such that the BCMCS_FLOW_ID can be derived. Steps 5a and 5b are unnecessary if the mapping between IP address/port and BCMCS_FLOW_ID is acquired via BCMCS Information Acquisition.

6. SGDD is sent over the broadcast channel from the BSD/A through BCMCS Content Server. The terminal tunes to the SG announcement channel and receives the SGDD(s). SGDD specifies the metadata for SGDU.

7. If the IP address and port number of the SG Delivery Channel, as provided by the SGDU metadata, is within the range of static mapping, the terminal will derive BCMCS_FLOW_ID from that information. Otherwise, the terminal can perform BCMCS Information Acquisition (not shown) to obtain the mapping between the IP address/port and BCMCS_FLOW_ID and other parameters.

8. SG/SGDU is sent over the broadcast channel from the BSD/A through BCMCS Content Server. The terminal tunes to the SG delivery channel, which transmits SGDUs, to receive the SG.

## 6.4 Generic adaptation related to OMA-TS-BCAST_SvcCntProtection and OMA-TS-DRM-XBS

### 6.4.1 Content Encryption

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-ServContProt], Sections 4.1 and 9).

### 6.4.2 Key Management

As defined by the generic BCAST Enabler specifications ([BCAST12-ServContProt] and [BCAST12-SG]). Specifically, key registration procedures SHALL comply with "Subscriber Key Establishment using a (R-)UIM/CSIM" procedures defined in [BCAST12-ServContProt], Section 6.5.2.

(Editor's Note: The "Subscriber Key Establishment using a (R-)UIM/CSIM" procedures shall pertain to either pre-provisioned key management as natively specified for BCMCS, or GBA for bootstrapped key derivation.)

#### 6.4.2.1 Signalling of Key Management Information

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-SG], Section 5.1.2.4 and [BCAST12-ServContProt], Section 10.)

### 6.4.3 Service and Content Protection for Stream and File Distribution

#### 6.4.3.1 DRM Profile

##### 6.4.3.1.1 Streams

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-ServContProt], Sections 4.1, 5.6.1, 5.7 and 5.9).

##### 6.4.3.1.2 Files

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-ServContProt], Sections 4.1, 5.6.2, 5.7 and 5.9).

#### 6.4.3.2 Smartcard Profile

It should be noted that Smartcard Profile based service and content protection functionality are not limited to BCAST Terminals containing a removable secure function or module known as the Smartcard. Such functionality could also be made available on UIM-equipped BCAST Terminals (UIM, or User Interface Module, corresponds to a non-removable secure function/module which is employed by (some) mobile terminals which operate according to 3GPP2 specifications).

##### 6.4.3.2.1 Streams

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-ServContProt], Sections 4.1, 6.6, 6.7, 6.8.1 and 6.9).

##### 6.4.3.2.2 Files

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-ServContProt], Sections 4.1,6.6, 6.7, 6.8.2 and 6.9).

## 6.5 Generic adaptation related to OMA-TS-BCAST-Distribution

When using the broadcast channel, the BCAST-enabled broadcast services can be delivered in the individual BCMCS programs. Various high level scenarios are possible as described below in Sections 6.5.1 and 6.5.2.

### 6.5.1 OMA BCAST-enabled Services delivered as separate BCMCS Programs

It SHALL be possible to provide BCAST-enabled services as defined by [BCAST12-SG] as separate BCMCS Programs as shown in Figure 2 below. In the example of the following call flow, the BCAST Service Guide is delivered as a BCMCS Program, and the service discovery and acquisition of the SG is identical to that described in Section 6.3.7.1 (and shown by

Figure 1). In addition, the SG indicates the following services are available for subscription: Basic TV, Basic Radio with interactivity, and Notification. Each of these services (as defined in OMA BCAST SG) is provided as a separate BCMCS Program. Upon their discovery, these services are subscribed by and then delivered to the corresponding terminal. Note: the service types as shown are purely for illustrative purposes, and do not imply implementation.
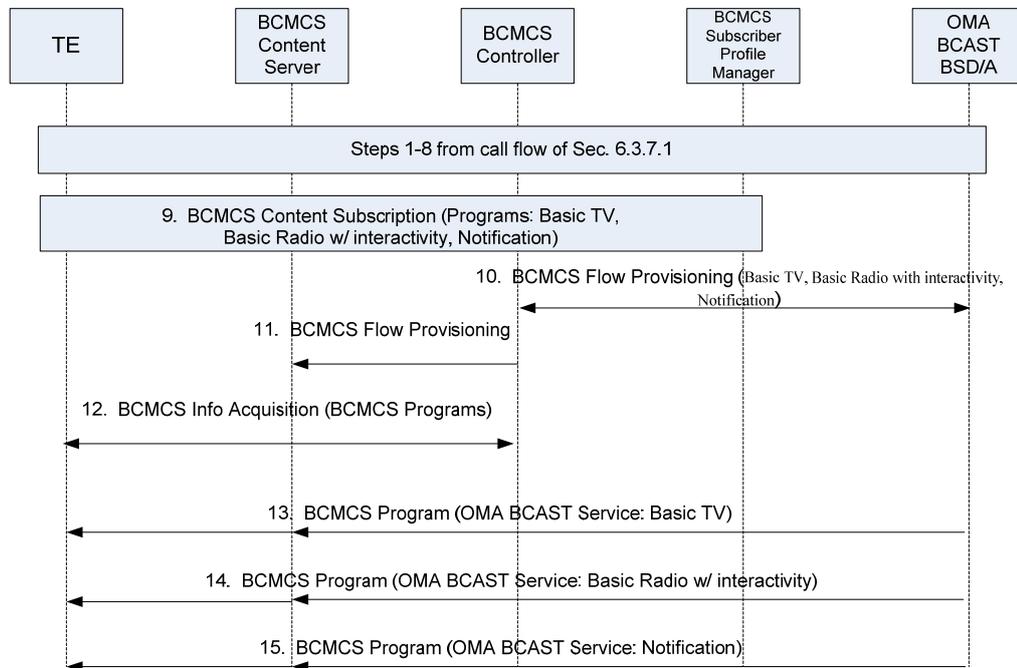


**Figure 2: OMA BCAST Services Delivered as Separate BCMCS Programs**

**Figure 2: OMA BCAST Services Delivered as Separate BCMCS Programs**

Steps 1-8: same as call flow shown in Figure 1 of Section 6.3.7.1.

9.  Upon receiving the SG, the user subscribes to the BCMCS Programs (Basic TV, Basic Radio with interactivity, and Notification) with the BCMCS Subscriber Profile Manager via out of band mechanisms.

10. The OMA BCAST BSD/A performs flow provisioning for the scheduled BCMCS Programs. In flow provisioning, the IP Address/ port, schedule, QoS for BCMCS programs are provided.

11. The BCMCS Controller performs flow provisioning with the BCMCS CS (see [3GPP2 X.S0022-A]).

12. The terminal performs BCMCS Information Acquisition to obtain mapping info and link layer info etc for the BCMCS Programs of interest.

Steps 13-15: at the scheduled times, the terminal receives the various BCMCS services as BCMCS programs.

## 6.5.2     Coexistence of OMA BCAST-enabled Services with BCMCS Native Services

The BCMCS native services and OMA BCAST-enabled broadcast services may coexist.  In this case, either BCMCS information acquisition is used for delivery of service information via Application Information (see [3GPP2 X.S0022-A]) or the BSD/A has program information for BCMCS native services so that integrated SG can be sent from BSD/A (see Section 6.3.7.1).  As shown in the call flow below, the integrated Service Guide is provided as a BCMCS Program from the BSD/A. Note: the services types as shown are purely for illustrative purposes, and do not imply implementation.
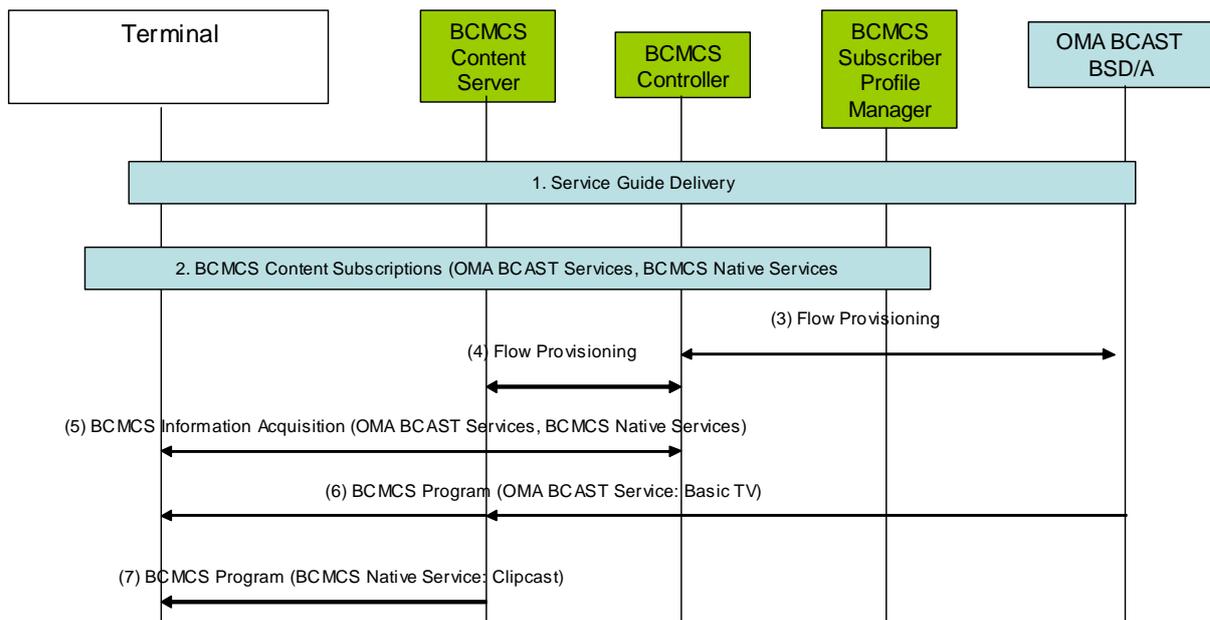


**Figure 3: Co-existence between BCAST-enabled and BCMCS Native Services**

1. The BSD/A sends service Guide including the program info for all OMA BCAST-enabled services and BCMCS native services as specified in Section 6.3.7.1.

2. Upon receiving the SG, the user subscribes to the BCMCS Programs (BCAST-related programs or BCMCS native programs) with the BCMCS Subscriber Profile Manager via out of band mechanisms.

3. The OMA BCAST BSD/A performs flow provisioning for the scheduled BCAST-related programs and BCMCS native programs.  In flow provisioning, the IP Address/ port, schedule, QoS for BCMCS programs are provided.

4. The BCMCS Controller performs flow provisioning including both BCAST-related programs and BCMCS native programs with the BCMCS CS (see [3GPP2 X.S0022-A]).

5. The terminal performs BCMCS Information Acquisition to obtain mapping info and link layer info etc for the all BCMCS Programs of interest.

6. In the scheduled time, the BSD/A sends BCAST-enabled services/content through the BCMCS Content server.

7. In the scheduled time, the BCMCS Content server sends BCMCS native programs.

## 6.5.3     File Distribution

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-Distribution]).  Note that currently, BCMCS does not natively specify file distribution services.

### 6.5.3.1        Associated Delivery Procedures

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-Distribution], Section 5.3).

### 6.5.3.2        File Distribution Network Interface between BCAST and BCMCS (FD-B1)

The BCMCS Content Server is the network entity that distributes BCMCS programs to terminals.  For adaptation to BCMCS, BCAST Terminals SHALL support ALC and MAY support FLUTE for file distribution as specified by [3GPP2 C.S0070-0]. Note that at this time file distribution services and associated architecture and protocols have not been specified in [3GPP2 X.S0022-A].  For flexibility of adaptation to BCMCS the BSD/A SHALL be configurable to support any one of the three protocol interface configurations for FD-B1 as shown in Sections 6.5.3.2.1 through 6.5.3.2.3.

Note: the protocol stacks spanned by X-3 and X-4 are outside the scope of the BCAST Enabler.

### 6.5.3.2.1        Configuration 1: File Object/(FTP or HTTP)/TCP/IP
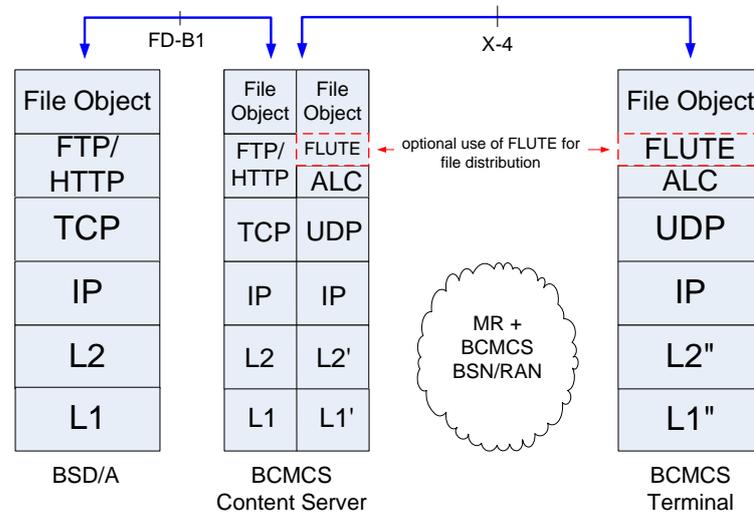


**Figure 4: FD-B1 Interface Configuration 1: File Object/(FTP/HTTP)/TCP/IP**

In the FD-B1 interface configuration shown in Figure 4, the BCMCS Content Server acts as a virtual content source.  It terminates the application layer file object received from the BSD/A, and acts as the final application level entity by manipulating (e.g. reformatting) that content prior to delivery to the BCMCS BSN/RAN.  Whereas unicast IP address is used over FD-B1, Multicast IP address and port, corresponding to the BCMCS Flow ID of the program, are used on the BDS side of the BCMCS Content Server.

### 6.5.3.2.2        Configuration 2: (FTP or HTTP)/TCP/IP
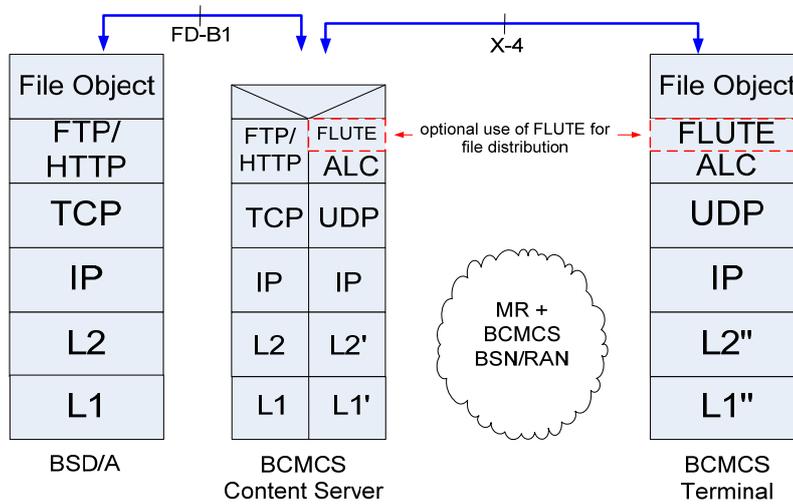


**Figure 5: FD-B1 Interface Configuration 2: (FTP or HTTP)/TCP/IP**

In the FD-B1 interface configuration shown in Figure 5, the BCMCS Content Server receives the file from the BSD/A using FTP/HTTP over TCP/IP.  It then distributes the file to the terminal using (FLUTE/)ALC.  Similar to Configuration 1, unicast IP address is used over FD-B1, whereas Multicast IP address and port are used on the BDS side of the BCMCS Content Server.

### 6.5.3.2.3        Configuration 3: IP in IP Tunneling
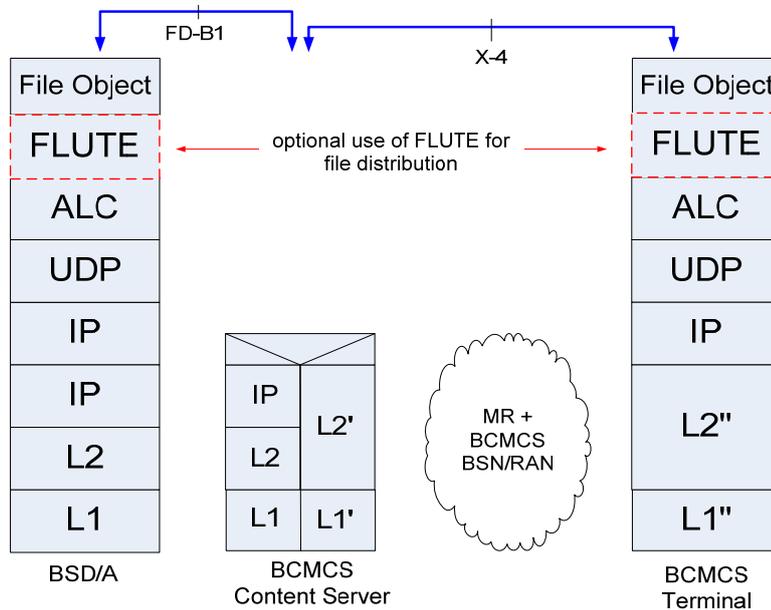


**Figure 6: FD-B1 Interface Configuration 3: IP in IP Tunnelling**

In the FD-B1 interface configuration shown in Figure 6, Multicast IP packets carrying application content over (FLUTE)/ALC/UDP are tunnelled in unicast IP packets destined for the BCMCS Content Server.  The BCMCS Content Server decapsulates the received IP packet by removing the outer IP packet header and relays the inner Multicast IP packet via the MR+BCMCS BSN/RAN to the terminal.  Similar to Configuration 2, this model may be useful to operators who want to keep the BCMCS Content Server functionality simple.

#### 6.5.3.2.4 BSD/A/Content Server to BCMCS Multicast Router Interface
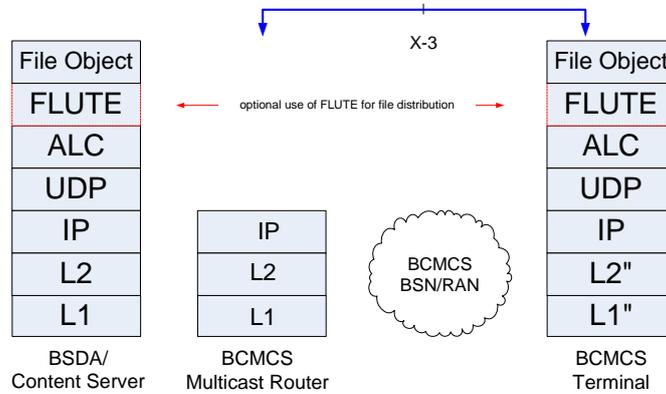


**Figure 7: BSD/A/Content Server to Multicast Router Interface Configuration**

As an implementation option shown in Figure 7, the BSD/A and the BCMCS Content Server functions may be combined as a single physical entity. This configuration may be desirable to an operator which owns both the BCAST Enabler and the BCMCS system, in which case it needs not deploy two boxes that perform essentially the same function. Note that in this case, FD-B1 does not exist.

### 6.5.4 Stream Distribution

As defined by the generic BCAST Enabler specifications (namely, [BCAST12-Distribution]).

#### 6.5.4.1 Stream Distribution Network Interface between BCAST and BCMCS (SD-B1)

The BCMCS Content Server provides stream delivery of BCMCS programs to terminals. As in OMA BCAST, RTP is specified as the streaming protocol in BCMCS. For flexibility of adaptation to BCMCS the BSD/A SHALL be configurable to support any one of the four protocol interface configurations for SD-B1, as shown in Sections 6.5.4.1.1 through 6.5.4.1.4.

Note: In this section, the protocol stacks spanned by X-3 and X-4 are outside the scope of the BCAST Enabler.

#### 6.5.4.1.1 Configuration 1: App/RTP/UDP/IP
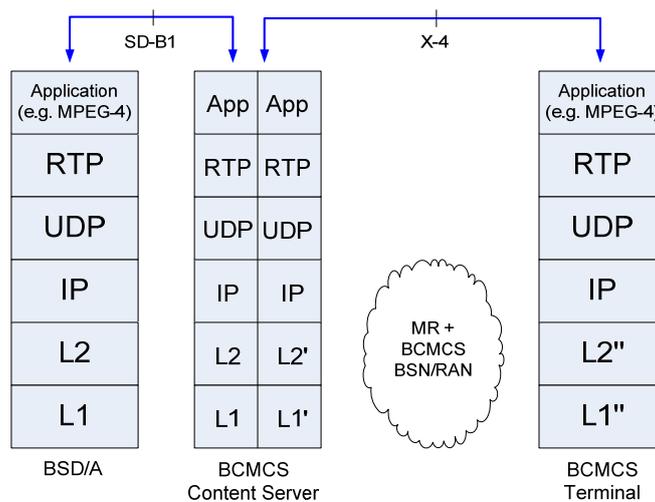


**Figure 8: SD-B1 Interface Configuration 1: App/RTP/UDP/IP**

In the SD-B1 interface configuration shown in Figure 8, the BCMCS Content Server acts as a virtual content source.  It terminates the stream application layer content received from the BSD/A, and acts as the final application level entity by manipulating (e.g. reformatting) that content prior to delivery to the BCMCS BSN/RAN.  Whereas unicast IP address is used over SD-B1, Multicast IP address and port, corresponding to the BCMCS Flow ID of the program, are used on the BDS side of the BCMCS Content Server.

### 6.5.4.1.2    Configuration 2: RTP/UDP/IP
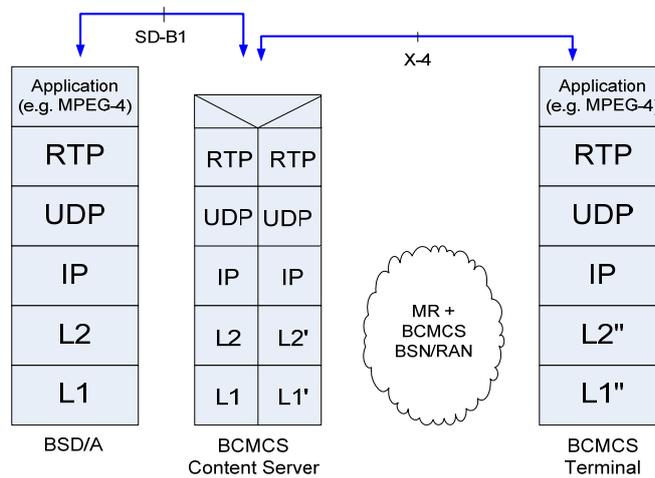


**Figure 9: SD-B1 Interface Configuration 2: RTP/UDP/IP**

In the SD-B1 interface configuration shown in Figure 9, the program content is streamed from the BSD/A to the BCMCS Content Server using RTP/UDP/IP.  The BCMCS Content Server subsequently initiates another RTP session for relaying the application content over the BCMCS RAN, to the terminal.  Similar to Configuration 1, unicast IP address is used over SD-B1, and Multicast IP address and port are used on the BDS side of the BCMCS Content Server.  This model is typically used if service protection as specified in [3GPP2 X.S0022-A] and [3GPP2 S.S0083-A] is performed by the BCMCS Content Server.

### 6.5.4.1.3    Configuration 3: UDP/IP
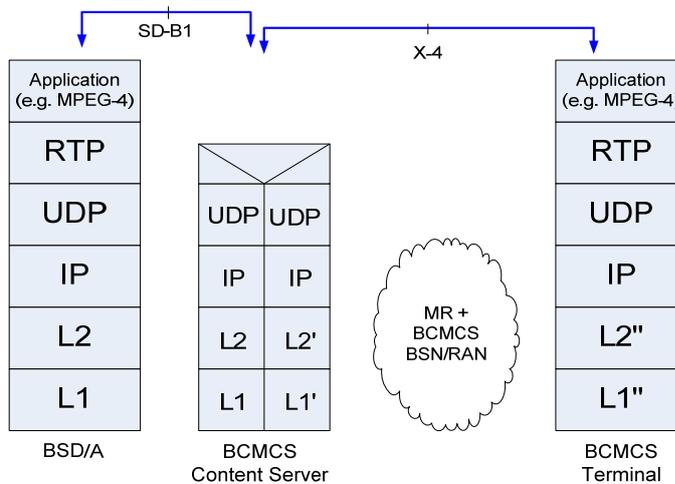


**Figure 10: SD-B1 Interface Configuration 3: UDP/IP**

In the SD-B1 interface configuration shown in Figure 10, the BCMCS Content Server relays RTP from the BSD/A to the terminal, and performs IP and UDP processing (i.e., BCMCS flow mapping).  Similar to Configuration 1, unicast IP address is used over SD-B1, and Multicast IP address and port are used on the BDS side of the BCMCS Content Server.  This model is applicable if service protection as specified in [3GPP2 X.S0022-A] and [3GPP2 S.S0083-A] is not performed by the BCMCS Content Server, and the operator wants to keep the BCMCS Content Server functionality simple.

### 6.5.4.1.4        Configuration 4: IP in IP Tunneling
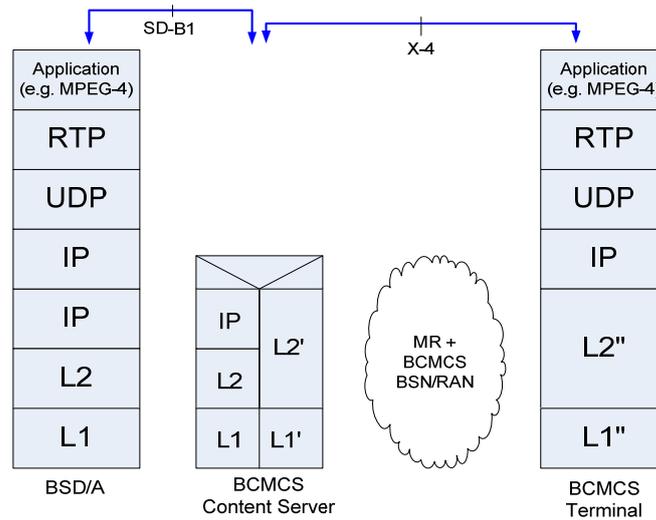


**Figure 11: SD-B1 Interface Configuration 4: IP in IP Tunneling**

In the SD-B1 interface configuration shown in Figure 11, Multicast IP packets carrying application content over RTP/UDP are tunnelled in unicast IP packets destined for the BCMCS Content Server.  The BCMCS Content Server decapsulates the received IP packet by removing the outer IP packet header and relays the inner Multicast IP packet via the MR+BCMCS BSN/RAN to the terminal.  Similar to Configuration 3, this model is valid if service protection as specified in [3GPP2 X.S0022-A] and [3GPP2 S.S0083-A] is not performed by the BCMCS Content Server, and the operator wants to keep the BCMCS Content Server functionality simple.

### 6.5.4.1.5          BSD/A/Content Server to BCMCS Multicast Router Interface
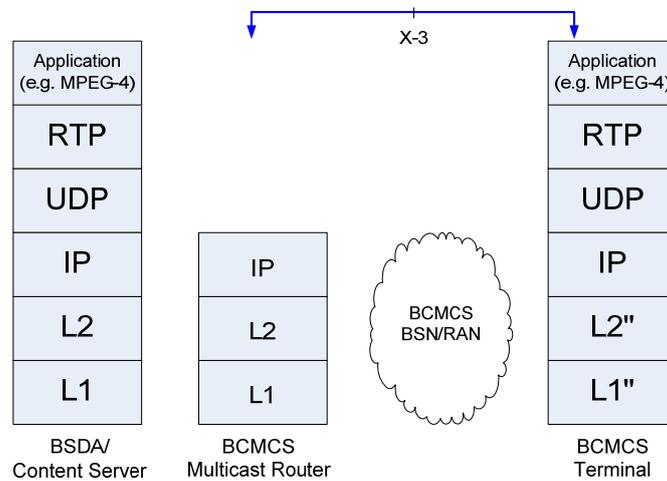


**Figure 12: BSD/A/Content Server to Multicast Router Interface Configuration**

As an implementation option, as shown in Figure 12, the BSD/A and the BCMCS Content Server functions may be combined as a single physical entity.  This configuration may be desirable to an operator which owns both the BCAST Enabler and the BCMCS system, in which case it needs not deploy two boxes that perform essentially the same function.  Note that in this case, SD-B1 does not exist.

## 6.5.5     Media Codecs and Formats

The BCAST Enabler does not define any media codecs and format, but references those specified in the underlying BDSs. Therefore, for BCMCS adaptation, BCAST Terminal support for media codecs SHALL comply with [3GPP2 C.S0070-0].

# 7. BDS-Specific Adaptation to BCMCS Functionality

This Section describes which BCAST technologies are chosen from BCMCS and how the 9 BCAST Functions are adapted for BCMCS network. The adaptation can be implemented via restrictions and extensions of the BCAST specifications (namely OMA-TS-BCAST-Services, OMA-TS-BCAST-ServiceGuide, OMA-TS-BCAST-SvcCntProtection, OMA-TS-BCAST-Distribution, and OMA-TS-DRM-XBS). The provisions in this section take precedence over the ones in the BCAST specifications to enable BCAST services using BCMCS adopted functionality to be distributed over BCMCS network allowing service sharing for BCMCS terminals.

BDS Specific adaptation MAY be supported by BCAST Network entities and SHALL be supported by BCAST Terminal.

All normative statements in this specification are only applicable in the case OMA BCAST services are distributed over BCMCS network.

Similar to Section 6, the message flows and architecture diagrams as referenced in this section presume logically separate BCAST and BCMCS network-side functions.  In deployment, BCAST and BCMCS network functions may be integrated into one or more physical entities, in which case some of the protocol interfaces and associated message flows are not applicable (i.e., those corresponding reference points are internal to the  network entity(entities) and don't correspond to physical interfaces).

Signaling of the generic adaptation mode is performed using service provisioning procedures as defined in [BCAST12-Services].

## 7.1    Access to the IP layer

Same as that defined in Section 6.1.

## 7.2    BDS-Specific adaptation related to OMA-TS-BCAST_Services

### 7.2.1    Interaction

Same as that defined in Section 6.2.1.

### 7.2.2    Service Provisioning

Same as that defined in Section 6.2.2.  Service provisioning is not specified in BCMCS specifications.

### 7.2.3    Terminal Provisioning

Same as that defined in Section 6.2.3.  Terminal provisioning is not specified in BCMCS specifications.

### 7.2.4    Notification

Same as that defined in Section 6.2.4.

## 7.3    BDS-Specific adaptation related to OMA-TS-BCAST_ServiceGuide

### 7.3.1    Service Guide Delivery over Broadcast Channel

Same as that defined in Section 6.3.1.

### 7.3.2    Service Guide Delivery over Interaction Channel

Same as that defined in Section 6.3.2.

### 7.3.3    Service Guide Encoding

Same as that defined in Section 6.3.3.

### 7.3.4    Session Description Information

Same as that defined in 6.3.4.

#### 7.3.4.1       Session Description parameters for broadcast streamed media sessions

Same as that defined in Section 6.3.4.1.

#### 7.3.4.2       Session Description parameters for broadcast file delivery sessions

Same as that defined in Section 6.3.4.2, including the session descriptors for both ALC and FLUTE sessions.

### 7.3.5    Service Guide Data Model

Same as that defined in Section 6.3.5.

### 7.3.6    CellTargetArea in BCMCS

Same as that defined in Section 6.3.6.

### 7.3.7    Service Guide Discovery

Same as in Section 6.3.7.

## 7.4    BDS-Specific adaptation related to OMA-TS-BCAST_SvcCntProtection and OMA-TS-DRM-XBS

### 7.4.1    Encryption

Only SRTP is defined in BCMCS specifications for transport level encryption.  Note that since service protection is specified by [BCAST12-ServContProt] as terminal-optional, SRTP support by the BCAST Terminal is mandatory if it supports service protection, in the adaptation to BCMCS.  Specifically, the SRTP configuration SHALL comply with the specifications in [3GPP2 X.S0022-A], i.e.,

- The BAK shall be used as the SRTP Master Key.

- The SK_RAND is 32 bits and shall be extended to 112 bits by left-padding with zeros to form the SRTP Master Salt.

- The Packet Index is determined according to [RFC 3711] (SRTP), Section 3.3.1.

- The Key Derivation Rate shall be set to zero. The Key Derivation Function shall be the AES in Counter Mode as specified in [RFC 3711], Section 4.1.1.

The Master Key Index (MKI) shall be used for distributing the SK_RAND. The MKI shall be 40 bits and consist of 4-bit reserved field, 4-bit BAK_ID and 32-bit SK_RAND.  The 32-bit ROC is included in every Rth packet in the auth tag field as specified in the [3GPP2 X.S0022-A].

### 7.4.2    Key Management

Same as that defined in Section 6.4.2, with the qualification that the "Subscription Key Establishment using a (R-)UIM/CSIM" procedure SHALL pertain strictly to pre-provisioned key management as natively specified for BCMCS..

#### 7.4.2.1       Signaling of Key Management Information

Same as that defined in Section 6.4.2.1, with the following exception.  The information for identification of parameters related to LTKM stream as defined in [BCAST12-ServContProt], Section 10 are not relevant for key management adaptation to BCMCS, and SHALL NOT be used.  BCMCS Information Acquisition SHALL be used by the terminal to obtain the SEK/PEK (i.e.  BAK in native BCMCS specifications).

## 7.4.3    Service and Content Protection for Stream and File Distribution

### 7.4.3.1      DRM Profile

#### 7.4.3.1.1        Streams

Same as that defined in Section 6.4.3.1.1 with the exception that the DRM Profile is not applicable for service protection of streams.  In other words, the terminal SHALL support native BCMCS service protection as described in Sections 7.4.1 and 7.4.2.

#### 7.4.3.1.2        Files

Same as that defined in Section 6.4.3.1.2.

### 7.4.3.2      Smartcard Profile

It should be noted that Smartcard Profile based service and content protection functionality are not limited to BCAST Terminals containing a removable secure function or module known as the Smartcard.  Such functionality could also be made available on UIM-equipped BCAST Terminals (UIM, or User Interface Module, corresponds to a non-removable secure function/module which is employed by (some) mobile terminals which operate according to 3GPP2 specifications).

#### 7.4.3.2.1        Streams

Same as that defined in Section 6.4.3.2.1, with the qualification that for service protection, the Smartcard Profile SHALL be profiled according to the native BCMCS configuration of SRTP as specified in Sections 7.4.1and 7.4.2.

#### 7.4.3.2.2        Files

Same as that defined in Section 6.4.3.2.2.

# 7.5    BDS-Specific adaptation related to OMA-TS-BCAST-Distribution

## 7.5.1    OMA BCAST-enabled Services delivered as separate BCMCS Programs

Same as that defined in Section 6.5.1.

## 7.5.2    Coexistence of OMA BCAST-enabled Services with BCMCS Native Services

Same as that defined in Section 6.5.2.

## 7.5.3    File Distribution

Same as that defined in Section 6.5.3, with the qualification that unless otherwise specified, the BSD/A SHALL use ALC for file distribution.

### 7.5.3.1      Associated Delivery Procedures

Same as that defined in Section 6.5.3.1.  Associated Delivery Procedures are not specified in BCMCS specifications

## 7.5.4    Stream Distribution

Same as that defined in Section 6.5.4.  The BCAST Terminal SHALL NOT send RTCP receiver reports.

## 7.5.5    Media Codecs and Formats

The BCAST terminal SHALL be able to receive, decode and render the data streams produced by the codecs and corresponding payload types that are specified as MANDATORY in [3GPP2 C.S0070-0].

The BCAST terminal SHOULD be able to receive, decode and render the data streams produced by the codecs and corresponding payload types that are RECOMMENDED in [3GPP2 C.S0070-0].

The BCAST terminal MAY be able to receive, decode and render the data streams produced by the codecs and corresponding payload types that are specified as OPTIONAL in [3GPP2 C.S0070-0].

# 8. Provisioning

## 8.1 Flow Provisioning Across BDS-1 Interface

The BCAST AD document shows the relationships among the BCAST logical entities (See Figure xx in AD document). In AD document, the BDS-1 is the interface between the OMA BCAST Service Distribution/Adaptation function and the BDS Service Distribution function in the underlying access/core network. In the case of BCMCS, the BDS Service Distribution entity corresponds to the BCMCS Controller/Content Server. Figure 13 shows how the BCMCS architecture fits within the OMA architecture
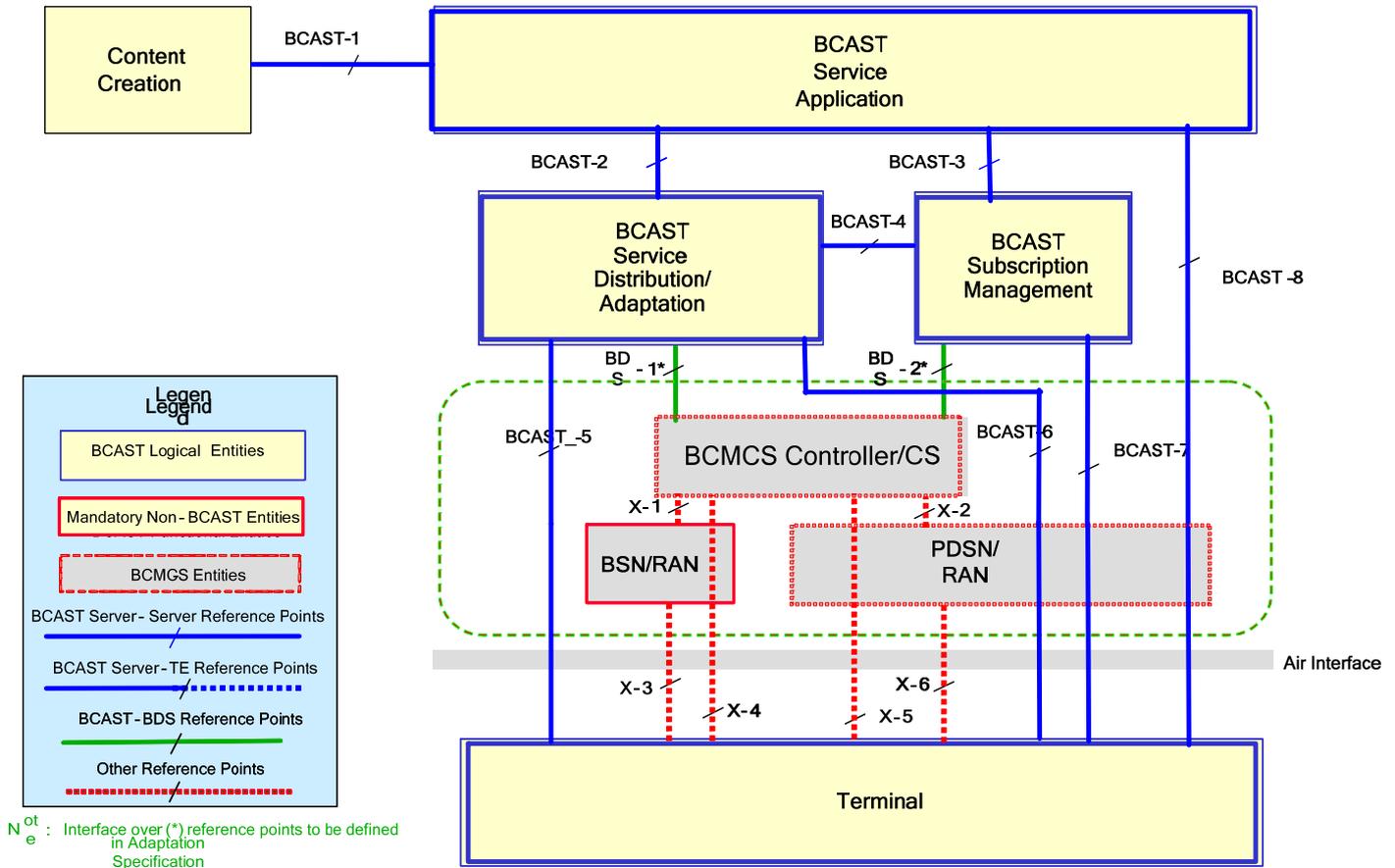


**Figure 13: BCMCS Adaptation Architecture Diagram**

The BDS-1 signalling interface between the BCMCS Controller and the BSD/A Function uses the *BSD/A-BCMCS Control Protocol*, as described in this section,  that provides a mechanism for BSD/A to provision BCMCS Flows (i.e., BCAST streams) on the BCMCS Controller and for BCMCS Controller to provide key to the BSD/A.

### 8.1.1 BCMCS Flow Life Cycle

When provisioning a BCMCS Flow for a multicast address/port the BCMCS Controller returns a BCMCS *Flow Handle* that the BSD/A uses to reference the flow.

A BCMCS Flow Handle references a unique BCMCS Flow provisioning on the BCMCS Controller and has a BCMCS Flow ID associated with it.   The same BCMCS Flow ID may be provisioned in multiple ways, as long as they do not overlap in time. Consequently, different BCMCS Flow Handles may have the same associated BCMCS Flow ID.

Each BCMCS Flow Handle shall be in one of the three states described below. Reception of messages or events, such as moving past the StartTime or EndTime of a BCMCS Flow, shall trigger transitions betweens these states.

A BCMCS Flow Handle is defined to be active at a point in time if it is assigned to a BCMCS Flow provisioned with an end time in the future. When the end time of a provisioned BCMCS Flow is passed, the BCMCS Flow Handle shall be returned to the inactive state. Inactive BCMCS Flow Handles are available for assignment to a new BCMCS Flow provisioning. The BSD/A may change the start and end times of a BCMCS Flow Handle. The BSD/A may remove a provisioned BCMCS Flow before scheduled end time. When a flow is removed, the BCMCS Flow Handle shall return to the Inactive state.
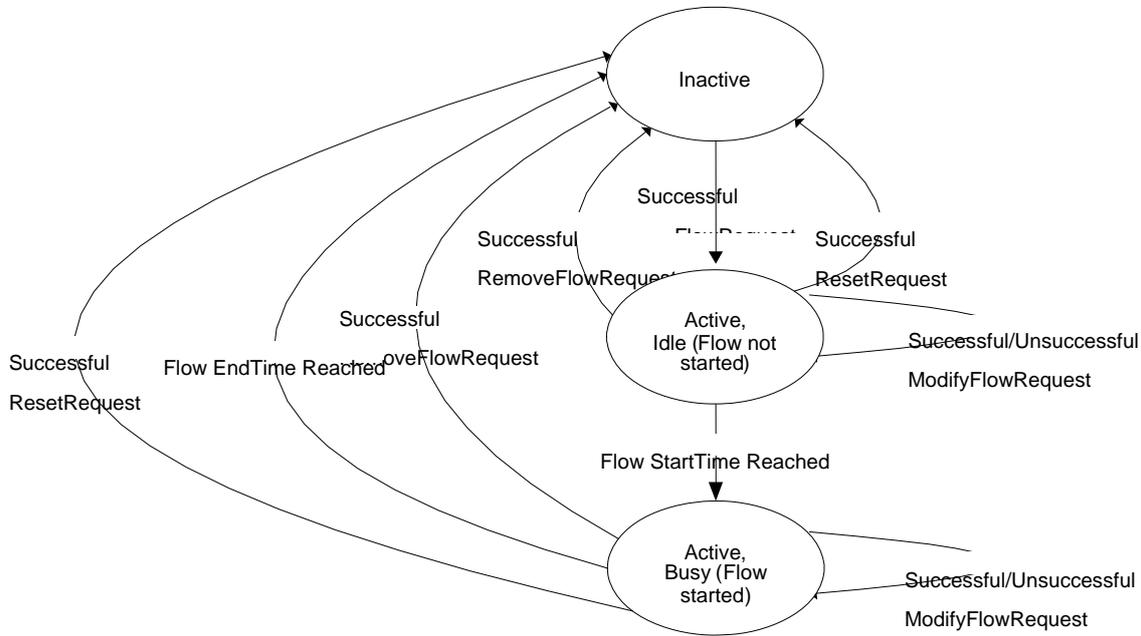


Note: Not all state transitions for unsuccessful cases are shown.

**Figure 14: BCMCS Flow Handle States**

### 8.1.1.1    Inactive State

This is the initial state of BCMCS Flow Handle. The BCMCS Controller shall maintain a pool of unassigned BCMCS Flow Handles to facilitate provisioning of BCMCS Flows. When a BCMCS Flow is created by a successful AddFlowRequest, the BCMCS Flow Handle shall transition to the {Active, Idle} state.

### 8.1.1.2    {Active, Idle} State

In this state the resources for corresponding BCMCS Flow are provisioned at the BCMCS controller but the BCMCS Flow is not actively transmitted from BSD/A. When a BCMCS Flow is removed by a successful RemoveFlowRequest, the BCMCS Flow Handle shall transition to the Inactive state and the BCMCS Flow shall be removed from the BCMCS Controller and BSD/A. A successful or unsuccessful ModifyFlowRequest shall maintain the BCMCS Flow Handle in the {Active, Idle} state. If a successful ResetRequest requires the provisioned BCMCS Flow Handle to be removed, the BCMCS Flow Handle shall transition to the Inactive state. If the start time for BCMCS Flow is reached, then the BCMCS Flow Handle shall transition to the {Active, Busy} state.

### 8.1.1.3 {Active, Busy} State

In this state, the corresponding BCMCS Flow is actively transmitted. When a BCMCS Flow is removed by a successful RemoveFlowRequest, the BCMCS Flow Handle shall transition to the Inactive state. The provisioned BCMCS Flow is removed from the BCMCS Controller and BSD/A. A successful or unsuccessful ModifyFlowRequest shall maintain the BCMCS Flow Handle in the {Active, Busy} state. If a successful ResetRequest requires the provisioned BCMCS Flow Handle to be removed, the BCMCS Flow Handle shall transition to the Inactive state. If the end time for the BCMCS Flow is reached, then the BCMCS Flow Handle shall transition to the Inactive state.

## 8.2 BSD/A-BCMCS Control Protocol

The BSD/A-BCMCS Control messages are sent over the TCP transport protocol. TCP port number SHALL be configurable on the BSD/A and BCMCS Controller.

### 8.2.1 BSD/A-BCMCS Control Protocol Messages

Table 2 lists the BSD/A-BCMCS Control Protocol message types.

| Message Type | Description |
|---|---|
| AddFlowRequest | Sent by the BSD/A to the BCMCS Controller to add one or more BCMCS Flows associated with one program. |
| AddFlowResponse | Sent by the BCMCS Controller to the BSD/A with result of AddFlowRequest message. |
| ModifyFlowRequest | Sent by the BSD/A to the BCMCS Controller to request modifying parameters of one or more provisioned BCMCS Flow(s). |
| ModifyFlowResponse | Sent by the BCMCS Controller to the BSD/A with result of ModifyFlowRequest message. |
| RemoveFlowRequest | Sent by the BSD/A to the BCMCS Controller to request removing one or more BCMCS Flow(s). |
| RemoveFlowResponse | Sent by the BCMCS Controller to the BSD/A with result of RemoveFlowRequest message. |
| ResetRequest | Sent by either BSD/A or BCMCS Controller to synchronize the flow state. |
| ResetResponse | Sent by either BSD/A or BCMCS Controller in response to the ResetRequest message. |
| RefreshKeyRequest | Sent by the BSD/A to the BCMCS Controller to request the BAK related to one or more existing Flow Handles(s). |
| RefreshKeyResponse | Sent by the BCMCS Controller to the BSD/A to provide the new BAK(s) related to Flow Handle(s), either in response to the RefreshKeyRequest message or autonomously sent before the BAK expiry time. |
| BCASTTransmissionAreaRequest | Sent by the BSD/A to the BCMCS Controller to indicate the broadcast transmission area(s) related to one or more existing Flow Handles(s). |
| BCASTTransmissionAreaResponse | Sent by the BCMCS Controller to the BSD/A with result of BCASTTransmissionAreaRequest message. |

**Table 2: BSD/A-BCMCS Control Protocol Message**

### 8.2.2 BSD/A-BCMCS Control Protocol Message Procedures

#### 8.2.2.1 Add BCMCS Flow

The AddFlowRequest message is sent by the BSD/A to the BCMCS Controller to request provisioning at the BCMCS Controller for one or more BCMCS Flows.

The BCMCS Controller can accept or deny the request through the AddFlowResponse message.

#### 8.2.2.1.1        BSD/A Functional Requirements

The BSD/A shall send an AddFlowRequest message to the BCMCS Controller to provision one or more BCMCS Flows associated with one program. The BSD/A shall set StartTime greater than current time and EndTime later than StartTime.

The BSD/A shall send an AddFlowRequest $T_{RECV}$ seconds[1] in advance before the start of the flow(s).

The BSD/A shall initiate sending content for a BCMCS Flow at the scheduled start time, if and only if the AddFlowRequest was successfully accepted by the BCMCS Controller.

#### 8.2.2.1.2        BCMCS Controller Functional Requirements

If the AddFlowRequest is accepted, the BCMCS Controller shall allocate a BCMCSFlowHandle for each flow and send an AddFlowResponse message with the ReasonCode IE set to SUCCESS to the BSD/A.

If the AddFlowRequest is accepted but SDPParameters received in AddFlowRequest includes a multicast IP address/port that is already used by other BCMCS flow, the BCMCS Controller shall allocate a new multicast IP address/port associated with the flow and returns it in the MulticastFlowAddress_BCMCSFlowHandle IE to the BSD/A with the ReasonCode IE set to SUCCESS. The BCMCS Controller is the central entity managing the multicast IP address/ports allocation to flows.

If the BCMCS Controller receives an AddFlowRequest with End Time earlier than Start Time for a particular flow, it shall reject that flow and shall set the ResultCode corresponding to that flow to INVALID_PARAMETER_VALUE and add the EndTime and StartTime IEIs (Information Element Identifiers, see Section 8.2.4.7), in the FailedParameter element for that flow.

If the BCMCS Controller receives an AddFlowRequest with parameters outside their defined range for a particular flow, it shall set the ResultCode corresponding to that flow to INVALID_PARAMETER_VALUE and add their IEIs in the FailedParameter element for that flow.

If the BCMCS Controller receives an AddFlowRequest with QoSParameter values for a particular flow that cannot be supported, it shall set the ResultCode corresponding to that flow to INVALID_PARAMETER_VALUE. The BCMCS Controller may include a supported QoSParameters for the corresponding flow and shall add the QoSParameter IEI (Information Element Identifier, see section 8.2.4.7) in the FailedParameter element for that flow in the AddFlowResponse message.

If the BCMCS Controller receives an AddFlowRequest with End Time earlier than Start Time for a particular flow, it shall reject that flow and shall set the ResultCode corresponding to that flow to INVALID_PARAMETER_VALUE and add the EndTime and StartTime IEIs (Information Element Identifiers, see section 8.2.4.7 ) in the FailedParameter element for that flow.

### 8.2.2.2        Modify BCMCS Flow

The ModifyFlowRequest message is sent by the BSD/A to the BCMCS Controller to modify start time, and/or end time of one or more BCMCS Flows provisioned through an AddFlowRequest message. Two types of modifications are allowed:

- Changing the start and/or end times.
- Extending the end time of a BCMCS Flow(s) in progress.

#### 8.2.2.2.1        BSD/A Functional Requirements

The BSD/A may send the ModifyFlowRequest message to the BCMCS Controller to change the start time, and/or end time of the flow(s) that have not started.

The BSD/A shall not send the ModifyFlowRequest message to change the start time of the flow(s) that have already started.

The BSD/A may send the ModifyFlowRequest message to the BCMCS Controller to change the end time of the flow(s) that have started.

In a ModifyFlowRequest message, if both StartTime and EndTime parameters are included, the BSD/A shall set the StartTime earlier than EndTime.

In a ModifyFlowRequest message, the BSD/A shall include at least one of the StartTime or EndTime parameters.

---

[1] The value of $T_{RECV}$ is an impletementation specific value or configurable by the operator

#### 8.2.2.2.2 BCMCS Controller Functional Requirements

The BCMCS Controller shall send a ModifyFlowResponse message to a BSD/A within $T_{resp}$ milliseconds[2] of receiving a ModifyFlowRequest message. The value of $T_{resp}$ shall be set to TBD.

If the BCMCS Controller receives a ModifyFlowRequest message with one of the following condition:

- StartTime later than EndTime, or

- StartTime earlier than the current time and the StartTime is not equal to the original start time of the BCMCS Flow,

the BCMCS Controller shall send the ModifyFlowResponse message with the ReasonCode IE set to INVALID_PARAMETER_VALUE and add the StartTime IE in the FailedParameter.

If the BCMCS Controller receives a ModifyFlowRequest message with a StartTime earlier than the current time and the Start Time is equal to the original start time of the BCMCS Flow, the BCMCS Controller shall regard the message to have a valid start time.

If the BCMCS Controller receives a ModifyFlowRequest message with an EndTime earlier than the current time, the CS shall send the ModifyFlowResponse message with the ReasonCode IE set to INVALID_PARAMETER_VALUE and add the EndTime IEI in the FailedParameter.

If the BCMCS Controller receives a ModifyFlowRequest message with a StartTime that cannot be supported due to resource limitations, the BCMCS Controller shall send the ModifyFlowResponse message with the ReasonCode IE set to RESOURCES_NOT_AVAILABLE and add the StartTime IEI in the FailedParameter.

If the BCMCS Controller receives a ModifyFlowRequest message with an EndTime that cannot be supported due to resource limitations the BCMCS Controller shall send the ModifyFlowResponse message with the ReasonCode IE set to RESOURCES_NOT_AVAILABLE and add the EndTime IEI in the FailedParameter.

If the BCMCS Controller indicates failure in the ModifyFlowResponse message then, the state and parameters associated with the BCMCSFlowHandle received in the ModifyFlowRequest shall remain the same as prior to the receipt of this message.

### 8.2.2.3 Remove BCMCS Flow

The RemoveFlowRequest message is sent by the BSD/A to the BCMCS Controller to remove provisioned BCMCS Flow(s). The Active BCMCS Flows that are either Idle or Busy can be removed. The BCMCS Controller sends the RemoveFlowResponse message to the BSD/A to indicate the result of the RemoveFlowRequest message.

#### 8.2.2.3.1 BSD/A Functional Requirements

The BSD/A may send the RemoveFlowRequest message to the BCMCS Controller to remove one or more BCMCS flows.

#### 8.2.2.3.2 BCMCS Controller Functional Requirements

If the BCMCS Controller accepts the RemoveFlowRequest message, it shall send the RemoveFlowResponse message to the BSD/A within $T_{resp}$[2] milliseconds of receiving the RemoveFlowRequest message, and with the ReasonCode IE set to SUCCESS to the BSD/A. The BCMCS Controller shall make the specified BCMCS Flow Handle inactive before sending the RemoveFlowResponse message. If the BCMCS Flow specified in the request is not in progress, then prior to sending the RemoveFlowResponse message, the BCMCS Controller shall release all state and resources associated with that flow.

If the BCMCS Controller receives a RemoveFlowRequest message including a BCMCS Flow Handle that is inactive, the BCMCS Controller shall return a RemoveFlowResponse message with the ResultCode set to INVALID_PARAMETER_VALUE for the corresponding BCMCS Flow Handle, and including the BCMCSFlowHandle IEI in the FailedParameter element.

---

[2] The value of $T_{resp}$ shall be set to implementation specific

### 8.2.2.4      State Synchronization

State Synchronization is achieved through the ResetRequest and ResetResponse messages.

The BSD/A or BCMCS Controller may send a ResetRequest message to synchronize the state with its peer. The ResetRequest message shall contain zero or more BCMCS Flow Handles that indicate that the sender (X) has state information for zero or more BCMCS Flow reservations. Upon receipt of the ResetRequest message, the receiver (Y) shall compare the received list of flow handles with its own list and removes state for any flow handles that are not included within the received message.

Y shall acknowledge the receipt of the ResetRequest with the ResetResponse message including its own remaining list of active Flow Handles. Upon receipt of the ResetResponse, X shall compare the received list of flow handles with its own list and removes state for any flow handles that are not included within the received message. At the end of the transaction for the ResetRequest/ResetResponse, X and Y have state for the same BCMCS Flow Handles (which is the common subset of original BCMCS Flow Handles present in X and Y).

The events that trigger the re-synchronization process at both entities (BCMCS Controller, BSD/A) are part of the local policy of BCMCS Controller and BSD/A, and not in the scope of the BDS-1 interface specification.

### 8.2.2.5      Refresh Keys

The RefreshKeyRequest message is sent by the BSD/A to the BCMCS Controller to request the Broadcast Access Key(s) (BAK) for one or more provisioned BCMCS Flows before the expiration time of the BAK(s).

The RefreshKeyResponse message is sent by the BCMCS Controller to the BSD/A to provide the new BAK(s) for a provisioned BCMCS Flow(s). The BCMCS Controller sends the new BAK to be used for any given flow.

The BCMCS Controller can accept or deny the request through the RefreshKeyResponse message.

#### 8.2.2.5.1        BSD/A Functional Requirements

The BSD/A shall send a RefreshKeyRequest message to the BCMCS Controller to request the new BAK for one or more provisioned BCMCS Flows, before the expiry time of the BAK has occurred.

#### 8.2.2.5.2        BCMCS Controller Functional Requirements

If the BCMCS Controller receives a RefreshKeyRequest message including an {Active, Idle} or {Active, Busy} BCMCS Flow Handle, the BCMCS Controller shall return the RefreshKeyResponse message to provide the associated BAKs for requested Flow Handles, if the request is successful. The BCMCS Controller shall set the ResultCode corresponding to that Flow Handle to SUCCESS.

If the BCMCS Controller receives a RefreshKeyRequest message with a BCMCSFlowHandle that is inactive, the BCMCS Controller shall return a RefreshKeyResponse message with the ResultCode corresponding to that flow set to INVALID_PARAMETER_VALUE, and including the BCMCSFlowHandle IEI in the FailedParameter element.

The BCMCS Controller may send two BAKs – current BAK to be used and a subsequent BAK to be used after the current one expires, for any given flow.

If the BAKParameter IE is included in the RefreshKeyResponse, the BCMCS Controller should use TLS-PSK [see RFC 4279] as a secure channel between the BCMCS Controller and BSD/A.

### 8.2.2.6      Broadcast Transmission Area

The BCASTTransmissionAreaRequest message is sent by the BSD/A to the BCMCS Controller to indicate the broadcast transmission areas associated with one or more Flow Handles.

The BCMCS Controller can accept or deny the request through the BCASTTransmissionAreaResponse message.

#### 8.2.2.6.1        BSD/A Functional Requirements

The BSD/A shall send a BCASTTransmissionAreaRequest message to the BCMCS Controller to indicate the broadcast transmission areas associated with one or more Flow Handles, before the end time of the BCMCS flow.

#### 8.2.2.6.2        BCMCS Controller Functional Requirements

The BCMCS Controller shall send a BCASTTransmissionAreaResponse message to the BSD/A after receiving a BCASTTransmissionAreaRequest message.

## 8.2.3    Message Format

A BSD/A-BCMCS control message consists of two parts: a Header and a Body. The header has a fixed length and a common format across messages as shown in.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Protocol Version = 01H | | | | | | | | 1 |
| Message Type = < 01H - 0CH > | | | | | | | | 2 |
| (MSB) | Message Length = < 0 - FF FFH > | | | | | | | 3 |
| | | | | | | | (LSB) | 4 |
| (MSB) | Transaction ID  = < 0 - FF FFH > | | | | | | | 5 |
| | | | | | | | (LSB) | 6 |
| (MSB) | Timestamp | | | | | | | 7 |
| | | | | | | | | … |
| | | | | | | | (LSB) | 14 |

**Table 3: Common Header Format and Body Format**

Protocol Version: This parameter indicates the version of the BSD/A-BCMCS Control Protocol the sender is using and it shall be set to 01H.

Message Type: The Message Type parameter identifies the type of message. The possible values for the Message Type parameter are listed in Table 4. Values 0H, 0BH – FFH are reserved.

| Value | Message Name |
|---|---|
| 01H | AddFlowRequest |
| 02H | AddFlowResponse |
| 03H | ModifyFlowRequest |
| 04H | ModifyFlowResponse |
| 05H | RemoveFlowRequest |
| 06H | RemoveFlowResponse |
| 07H | ResetRequest |
| 08H | ResetResponse |
| 09H | RefreshKeyRequest |
| 0AH | RefreshKeyResponse |
| 80H | BCASTTransmissionAreaRequest |
| 81H | BCASTTransmissionAreaResponse |
| Other Values | Reserved |

**Table 4: Message Type Values**

Message Length: 16-bit unsigned integer value that indicates the length of the message including the header fields and the body.

Transaction ID: 16-bit unsigned integer value that uniquely identifies a BSD/A-BCMCS Controller protocol transaction. This parameter is used by the sender to match a request with the corresponding response from the receiver. The sender increments this parameter by 1 for every new transaction with the receiver and the parameter wraps around at 65535 (FFFFH). The receiver copies the Transaction ID from the request into the response message.

Timestamp: 64-bit unsigned integer value in NTP format (specified in RFC 1305), indicating the time the message was created. The parameter is used to protect against replay attacks in conjunction with the mandatory presence of the AuthenticationExtension information element, which provides message authentication and integrity protection.

The body is of variable length and has Information Elements specific to each message type. The Information Elements are in TLV (Type, Length, and Value) format and can be Mandatory, Conditional, or Optional, as described in Table 5.

| Information Element | Description |
|---|---|
| **Mandatory (M)** | The Information Element is required for the message. |
| **Conditional (C)** | The Information Element is required in situations where a condition is met (the condition is given in the Description in the message definition). |
| **Optional (O)** | The Information Element is provided at the discretion of the implementation. |

**Table 5: Definition of Mandatory, Conditional, Optional**

## 8.2.3.1     AddFlowRequest

The message format of the AddFlowRequest is described in Table 6.

| Information Element | Reference | M/C/O |
|---|---|---|
| One occurrence of the following element: | | |
| ContentProviderID | See 8.2.4.14 | M |
| ProgramName | See 8.2.4.17 | M |
| StartTime | See 8.2.4.12 | M |
| EndTime | See 8.2.4.6 | M |
| ContentTunnelProtocolOption | See 8.2.4.4 | M |
| L3TunnelSourceAddress | See 8.2.4.9 | Ca |
| SDPParameters | See 8.2.4.15 | M |
| N (N=number of flows being added)  occurrences of the following element: | | |
| QoSParameters | See 8.2.4.13 | O |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

a Included only if 'ContentTunnelProtocolOption' = 0H (L3 Tunneling).

**Table 6: AddFlowRequest**

The feasible QoSParameters values depend on the characteristics of the content and the Wireless service Provider's network. The BSD/A requests an operating point for these parameters and the BCMCS Controller accepts the parameters if it is feasible.

### 8.2.3.2    AddFlowResponse

The message format of the AddFlowResponse is described in Table 7.

| Information Element | Reference | M/C/O |
|---|---|---|
| One occurrence of the following element: | | |
| L3TunnelDestinationAddress | See 8.2.4.8 | Cc |
| SDPParameters | See 8.2.4.15 | O |
| DelayOffset | See 8.2.4.5 | O |
| N occurrences of the following elements: | | |
| ResultCode | See 8.2.4.11 | M |
| MulticastFlowAddress_BCMCSFlow Handle | See 8.2.4.10 | Ca |
| QoSParameters | See 8.2.4.13 | O |
| FailedParameter | See 8.2.4.7 | Cd |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

a  Included only if 'ResultCode' = SUCCESS (0H).

c  Included if 'ResultCode' = SUCCESS (0H) and 'ContentTunnelProtocolOption' = 0H (L3 Tunnel).

d Included if 'ResultCode' is set to one of the following values: UNSUPPORTED_PARAMETER, POORLY_FORMED_PARAMETER, INVALID_PARAMETER_VALUE and MISSING_PARAMETER

**Table 7: AddFlowResponse**

### 8.2.3.3    ModifyFlowRequest

The message format of the ModifyFlowRequest is described in Table 8.

| Information Element | Reference | M/C/O |
|---|---|---|
| One occurrence of the following element: | | |
| StartTime | See 8.2.4.12 | Oa |
| EndTime | See 8.2.4.6 | Oa |
| N occurrences of the following elements: | | |
| BCMCSFlowHandle3 | See 8.2.4.3 | M |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

**Table 8: ModifyFlowRequest**

[a] At least one of the StartTime or EndTime shall be included.

---

[3] The BSD/A chooses BCMCSFlowHandle(s) associated with one BCMCS program.

### 8.2.3.4        ModifyFlowResponse

The message format of the ModifyFlowResponse is described in Table 9.

| Information Element | Reference | M/C/O |
|---|---|---|
| N occurrences (N=number of flows modified) of the following elements: | | |
| ResultCode | See 8.2.4.11 | M |
| FailedParameter | See 8.2.4.7 | C[a] |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

[a] Included if 'ResultCode' is set to one of the following values:
UNSUPPORTED_PARAMETER, POORLY_FORMED_PARAMETER,
INVALID_PARAMETER_VALUE and MISSING_PARAMETER

**Table 9: ModifyFlowResponse**

### 8.2.3.5        RemoveFlowRequest

The message format of the RemoveFlowRequest is described in Table 10.

| Information Element | Reference | M/C/O |
|---|---|---|
| N occurrences (N=number of flows removed) of the following elements: | | |
| BCMCSFlowHandle | See 8.2.4.3 | M |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

**Table 10: RemoveFlowRequest**

### 8.2.3.6        RemoveFlowResponse

The message format of the RemoveFlowResponse is described in Table 11.

| Information Element | Reference | M/C/O |
|---|---|---|
| N occurrences of the following elements: | | |
| ResultCode | See 8.2.4.11 | M |
| FailedParameter | See 8.2.4.7 | Ca |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

a                       Included if 'ResultCode' is set to one of the following values:
UNSUPPORTED_PARAMETER, POORLY_FORMED_PARAMETER,
INVALID_PARAMETER_VALUE and MISSING_PARAMETER

**Table 11: RemoveFlowResponse**

### 8.2.3.7      RefreshKeyRequest

The message format of the RefreshKeyRequest is described in Table 12.

| Information Element | Reference | M/C/O |
|---|---|---|
| N occurrences of the following elements: | | |
| BCMCSFlowHandle | See Section 8.2.4.3 | M |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

**Table 12: RefreshKeyRequest**

### 8.2.3.8      RefreshKeyResponse

The message format of the RefreshKeyRequest is described in Table 13.

Note that the BCMCS Controller may choose to return both the current as well as the subsequent BAK for an associated flow.

| Information Element | Reference | M/C/O |
|---|---|---|
| N occurrences of the following elements: | | |
| ResultCode | See 8.2.4.11 | M |
| BAKParameter | See 8.2.4.2 | Ca |
| FailedParameter | See 8.2.4.7 | Cb |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

a                        Included only if 'ResultCode' = SUCCESS (0H).
b                        Included if 'ResultCode' is set to one of the following values:
UNSUPPORTED_PARAMETER, POORLY_FORMED_PARAMETER,
INVALID_PARAMETER_VALUE and MISSING_PARAMETER

**Table 13: RefreshKeyResponse**

### 8.2.3.9      BCASTTranmsissionAreaRequest

The message format of the BCASTTransmissionAreaRequest is described in Table 14.

| Information Element | Reference | M/C/O |
|---|---|---|
| Location Area | See 8.2.4.16 | M |
| N (N=number of flows being sent/stopped in the above specified Location Area) occurrences of the following elements: | | |
| BCMCSFlowHandle | See 8.2.4.3 | M |
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

**Table 14: BCASTTransmissionAreaRequest**

### 8.2.3.10      BCASTTranmsissionAreaResponse

The message format of the BCASTTransmissionAreaResonse is described in Table 15.

| Information Element | Reference | M/C/O |
|---|---|---|
| ResultCode | See 8.2.4.11 | Ca |
| Location Area | See 8.2.4.16 | O |
| N (N=number of flows being sent/stopped in the above specified Location Area) occurrences of the following elements: | | |
| BCMCSFlowHandle | See 8.2.4.3 | O |

| Information Element | Reference | M/C/O |
|---|---|---|
| One occurrence of the following element: | | |
| AuthenticationExtension | See 8.2.4.1 | M |

a  Included only if all request in Requests are successful and 'ResultCode' = SUCCESS (0H).

**Table 15: BCASTTransmissionAreaResponse**

## 8.2.4  Information Elements

This section contains the coding and definition of the Information Elements (IEs) used in the messages defined in Section 8.2.3.

The following table contains a list of all the information elements used on the BSD/A-BCMCS control protocol. The table is sorted by the Information Element Identifier (IEI) coding which distinguishes one information element from another. The table also includes a reference to the section where the element coding can be found.

| Element Name | Identifier (IEI) | Section Reference |
|---|---|---|
| ResultCode | 01H | 8.2.4.11 |
| MulticastFlowAddress_ BCMCSFlowHandle | 02H | 8.2.4.10 |
| StartTime | 03H | 8.2.4.12 |
| EndTime | 04H | 8.2.4.6 |
| ContentProviderID | 05H | 8.2.4.14 |
| ContentTunnelProtocolOption | 06H | 8.2.4.4 |
| L3TunnelSourceAddress | 07H | 8.2.4.9 |
| BCMCSFlowHandle | 08H | 8.2.4.3 |
| L3TunnelDestinationAddress | 09H | 8.2.4.8 |
| DelayOffset | 0AH | 8.2.4.5 |
| FailedParameter | 0BH | 8.2.4.7 |
| AuthenticationExtension | 0CH | 8.2.4.1 |
| BAKParameters | 0DH | 8.2.4.2 |
| SDPParameters | 0EH | 8.2.4.15 |
| QoSParameters | 0FH | 8.2.4.13 |
| Location Area | 80H | 8.2.4.16 |
| ProgramName | 81H | 8.2.4.17 |
| All other values are reserved. | | |

**Table 16: Information Element Identifiers**

The rest of this section presents the definition of the BSD/A-BCMCS Control Protocol information elements.

### 8.2.4.1  AuthenticationExtension

The Authentication Extension shall be included in all BSD/A-BCMCS Control Protocol messages. This Authentication Extension information element marks the end of the authenticated data in the BSD/A-BCMCS Control Protocol messages. The structure of the extension is as shown in Table 17.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 0CH | | | | | | | | 1 |
| Length = < 0 - FFH > | | | | | | | | 2 |
| (MSB) | SPI | | | | | | | 3 |
| | | | | | | | | 4 |

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 5 |
| | | | | | | | (LSB) | 6 |
| (MSB) | | Authentication Data | | | | | | 7 |
| | | | | | | | | ... |
| | | | | | | | (LSB) | Variable |

**Table 17: AuthenticationExtension**

IEI: Information Element Identifier. It shall be set to 0CH.

Length: This field indicates the number of octets in this including IEI and Length fields. This field is set to 6 plus the number of bytes in the authenticator.

SPI: This four octet field is set to the Security Parameter Index as specified in [RFC 3344].

Authentication Data: For message authentication, the Authenticator field is calculated over TCP payload data except for Authenticator field, as specified in section 3.5.1 of [RFC 3344].

### 8.2.4.2 BAKParameters

The BCMCS Controller sends the BAK, BAK_ID, and BAK_Expire field to the BSD/A (in the given order). The structure of the BAKParameters is as shown in Table 18.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 0DH | | | | | | | | 1 |
| Length | | | | | | | | 2 |
| (MSB) | | Identifier Type | | | | | | 3 |
| Identifier | | | | | | | | N |
| (MSB) | | Reserved | | | BAK_ID | | | N+1 |
| | | | | | | | | N+2 |
| BAK | | | | | | | | ... |
| | | | | | | | | N+17 |
| | | | | | | | | N+18 |
| BAK_Expire | | | | | | | | ... |
| | | | | | | | | N+21 |

**Table 18: BAKParameters**

**IEI**: Information Element Identifier. It shall be set to 0DH.

**Length**: This field indicates the number of octets in this element including IEI and Length fields.

**Identifier Type**: This field shall be set to 00H if the Identifier field contains the BCMCSFlowHandler value. This filed shall be set to 04H if the Identifier field contains IPv4 Multicast Address and Port number. This filed shall be set to 06H if the Identifier field contains IPv6 Multicast Address and Port number. Other values are reserved. The sender shall set this field to 00H if the BCMCSFlowHandler is available.

**Identifier**: This field shall be set to the BCMCSFlowHandler value (4 octets) if the Identifier type field is set to 00H. This field shall be set to IPv4 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 04H. This filed shall be set to IPv6 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 06H.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| (MSB) | | Port Number = < C0 00H - C0 07H> | | | | | | 1 |
| | | | | | | | (LSB) | 2 |
| (MSB) | | IP Address (IPv4 or IPv6) | | | | | | 3 |

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | **...** |
| | | | | | | | (LSB) | **6 or 18** |

**Table 19: Field format for Port number and IP address**

**BAK_ID:** BAK identifier. A sequence number that identifies which value of BAK is currently valid for a particular BCMCS Multicast IP Flow; that is, the BAK is identified by the combination of a BCMCS_FLOW_ID and BAK_ID. For a particular BAK, the corresponding value of BAK_ID is the same for all users.

**BAK** 128-bit Broadcast Access Key: Provides access to one or more multicast IP flows of a particular set of BCMCS programs for a certain amount of time (for example, one day, week or month). Each encrypted set of BCMCS programs have a different BAK value.

**BAK_Expiry:** Indicates when the BAK will expire. This field shall be encoded using an unsigned integer and contain the number of seconds since January 1, 1970 00:00 UTC.

### 8.2.4.3 BCMCSFlowHandle

The BCMCS Flow Handle references a unique BCMCS Flow reservation. The structure of the BCMCSFlowHandle is as shown in Table 20.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 08H | | | | | | | | **1** |
| Length = 06H | | | | | | | | **2** |
| (MSB) | Value = < 0 - FF FF FF FFH > | | | | | | | **3** |
| | | | | | | | | **4** |
| | | | | | | | | **5** |
| | | | | | | | (LSB) | **6** |

**Table 20: BCMCSFlowHandle**

**IEI:** Information Element Identifier. It shall be set to 08H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields. This field shall be set to 6.

**Value**: 32-bit unsigned integer value indicating the BCMCS Flow Handle.

### 8.2.4.4 ContentTunnelProtocolOption

The ContentTunnelProtocolOption indicates what bearer protocol is used between BSD/A and BCMCS CS to transport content for the BCMCS Flow being added. The structure of the ContentTunnelProtocolOption is as shown in Table 21.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 06H | | | | | | | | **1** |
| Length = 03H | | | | | | | | **2** |
| Value = < 00H-FFH > | | | | | | | | **3** |

**Table 21: ContentTunnelProtocolOption**

**IEI:** Information Element Identifier. It shall be set to 06H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields. This field shall be set to 3.

Value: This 8-bit enumerated value identifies the type of bearer transport protocol used between the BSD/A and BCMCS Controller. The supported 'ContentTunnelProtocolOption' values are listed in Table 22 below:

| Value | Content Transport Protocol |
|---|---|
| 00H | L3 Tunnel |
| 01H-FFH | Reserved |

**Table 22: ContentTunnelProtocolOption Values**

### 8.2.4.5 DelayOffset

This information element indicates that the BSD/A shall start transmission of content for the BCMCS Flow at (StartTime – DelayOffset). This is to compensate for processing and transmission delays on the BSD/A-BCMCS CS link.  The structure of the DelayOffset is as shown in Table 23.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 0AH | | | | | | | | 1 |
| Length = 04H | | | | | | | | 2 |
| (MSB) | Value = < 0 - FF FFH > | | | | | | | 3 |
| | | | | | | | (LSB) | 4 |

**Table 23: DelayOffset**

**IEI:** Information Element Identifier. It shall be set to 0AH.

**Length:** This field indicates the number of octets in this element including IEI and Length fields. This field shall be set to 4.

Value: 16-bit unsigned integer value indicating time in milliseconds.

### 8.2.4.6 EndTime

This parameter indicates the end time of a BCMCS Flow.  The structure of the EndTime is as shown in Table 24.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 04H | | | | | | | | 1 |
| Length = 0AH | | | | | | | | 2 |
| (MSB) | Value = < 0 - FF FF FF FF FF FF FF FFH > | | | | | | | 3 |
| | | | | | | | | ... |
| | | | | | | | (LSB) | 10 |

**Table 24: EndTime**

**IEI:** Information Element Identifier. It shall be set to 04H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields. This field shall be set to 0A.

**Value**: 64-bit unsigned integer value in NTP format, as specified in [RFC 1305].

### 8.2.4.7 FailedParameter

This information element is included in response messages that have the ResultCode set to one of the following values: UNSUPPORTED_PARAMETER, POORLY_FORMED_PARAMETER, INVALID_PARAMETER_VALUE, and MISSING_PARAMETER.  The structure of the FailedParameter is as shown in Table 25.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 0BH | | | | | | | | 1 |
| Length = <variable> | | | | | | | | 2 |
| Number of Failed IEIs | | | | | | | | 3 |
| **Number of Failed IEIs occurrences of the following field:** | | | | | | | | |
| Identifier Type | | | | | | | | 4 |
| Identifier | | | | | | | | Variable N |
| Failed Information Element Identifier | | | | | | | | N+1 |

**Table 25: FailedParameter**

**IEI:** Information Element Identifier. It shall be set to 0BH.

**Length:** This field shall be set the number of Failed Information Element Identifiers in this element plus 2.

**Number of FaildIEIs:** This field shall be set number of Faild IEIs in this IE

**Identifier Type:** This field shall be set to 00H if the Identifier field contains the BCMCSFlowHandler value. This field shall be set to 04H if the Identifier field contains IPv4 Multicast Address and Port number. This field shall be set to 06H if the Identifier field contains IPv6 Multicast Address and Port number. The sender shall set this field to 00H if the BCMCSFlowHandler is available.

**Identifier:** This field shall be set to the BCMCSFlowHandler value (4 octets) associated with the FaildParametrs if the Identifier Type field is set to 00H. This filed shall be set to IPv4 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 04H. This filed shall be set to IPv6 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 06H.

**Failed Information Element Identifier**: This field contains a Failed Information Element Identifier.

### 8.2.4.8    L3TunnelDestinationAddress

This information element specifies the IP address of the BCMCS CS entity that is the destination endpoint of the L3 tunnel for a BCMCS Flow.  The structure of the L3TunnelDestinationAddress is as shown in Table 26.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|-------|
| Information Element Identifier (Type) = 09H | | | | | | | | 1 |
| Length = < 07H, 13H > | | | | | | | | 2 |
| IPVersion = < 04H, 06H > | | | | | | | | 3 |
| (MSB) | IPAddress | | | | | | | 4 |
| | | | | | | | | ... |
| | | | | | | | (LSB) | 7 or 19 |

**Table 26: L3TunnelDestinationAddress**

**IEI:** Information Element Identifier. It shall be set to 09H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields. It shall be set to 07H if the 'IP Version' field is set to 04H, or 13H if the 'IP Version' field is set to 06H.

**IP Version**: This parameter indicates the version of the IP protocol used as a Delivery Protocol for L3 Tunnelling. The supported 'IP Version' values are listed in Table 27 below:

| Value | IP Version |
|-------|-----------|
| 04H | IPv4 |
| 06H | IPv6 |

**Table 27: IP Version Values**

**IP Address:** 32-bit IPv4, or 128-bit IPv6 address, depending on the value of the 'IP Version' field.

### 8.2.4.9    L3TunnelSourceAddress

This information element specifies the IP address of the BSD/A entity that is the source endpoint of the L3 tunnel for a BCMCS Flow.  The structure of the L3TunnelSourceAddress is as shown in Table 28.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|-------|
| Information Element Identifier (Type) = 07H | | | | | | | | 1 |
| Length = < 07H, 13H > | | | | | | | | 2 |
| IP Version = < 04H, 06H > | | | | | | | | 3 |
| (MSB) | IP Address | | | | | | | 4 |
| | | | | | | | | ... |
| | | | | | | | (LSB) | 7 or 19 |

**Table 28: L3TunnelSourceAddress**

**IEI:** Information Element Identifier. It shall be set to 07H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields. It is set to 07H if the 'IP Version' field is set to 04H, or 13H if the 'IP Version' field is set to 06H.

**IP Version:** This parameter indicates the version of the IP protocol used as a Delivery Protocol for L3. The supported 'IP Version' values are listed in Table 27.

**IP Address:** 32-bit IPv4, or 128-bit IPv6 address, depending on the value of the 'IP Version' field.

### 8.2.4.10    MulticastFlowAddress_BCMCSFlowHandle

This Information Element specifies the IP multicast address and port and BCMCS FLOW Handle on which the BCMCS Flow is sent.  The structure of the MulticastFlowAddress is as shown in Table 29.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 02H | | | | | | | | 1 |
| Length = < 0DH, 19H > | | | | | | | | 2 |
| (MSB) | Port Number = < C0 00H - C0 07H> | | | | | | | 3 |
| | | | | | | | (LSB) | 4 |
| IP Version = < 04H, 06H > | | | | | | | | 5 |
| (MSB) | IP Address | | | | | | | 6 |
| | | | | | | | | … |
| | | | | | | | (LSB) | 9 or 21 |
| (MSB) | | | | | | | | 10 or 22 |
| BCMCSFlowHandle | | | | | | | | 11 or 23 |
| | | | | | | | | 12 or 24 |
| | | | | | | | (LSB) | 13 or 25 |

**Table 29: MulticastFlowAddress**

**IEI:** Information Element Identifier. It shall be set to 02H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields. It is set to 13H if the 'IP Version' field is set to 04H, or 25H if the 'IP Version' field is set to 06H.

**Port Number**: 32-bit port number. The range of permissible values is from 49152 (C000H) to 49159 (C007H).

**IP Version**: This parameter indicates the version of the IP protocol used as a Delivery Protocol for L3 Tunneling. The supported 'IP Version' values are listed in Table 27.

**IP Address:** 32-bit IPv4, or 128-bit IPv6 address, depending on the value of the 'IP Version' field.

**BCMCSFlowHandle:** 32-bit unsigned integer value indicating the BCMCS Flow Handle.

### 8.2.4.11    ResultCode

This information element is used to indicate the reason for occurrence of a particular event.  The structure of the ResultCode is as shown in Table 30.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 01H | | | | | | | | 1 |

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Length ||||||||| 2 |
| Identifier Type ||||||||| 3 |
| Identifier ||||||||| Variable N |
| Value = < 0 - FFH > ||||||||| N+1 |

**Table 30: ResultCode**

**IEI:** Information Element Identifier. It shall be set to 01H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields.

**Identifier Type:** This field shall be set to 00H if the Identifier field contains the BCMCSFlowHandler value. This filed shall be set to 04H if the Identifier field contains IPv4 Multicast Address and Port number. This filed shall be set to 06H if the Identifier field contains IPv6 Multicast Address and Port number. The sender shall set this field to 00H if the BCMCSFlowHandler is available.

**Identifier:** This field shall be set to the BCMCSFlowHandler value (4 octets) associated with the ReasonCode if the Identifier type field is set to 00H. This filed shall be set to IPv4 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 04H. This filed shall be set to IPv6 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 04H.

**Value**: 8-bit unsigned integer value indicating the processing result of a message request, as shown in Table 31.

| Value | Mnemonic | Description |
|---|---|---|
| 00H | SUCCESS | The Request was successfully completed |
| 01H | RESOURCES_NOT_AVAILABLE | Resources are not available at the time of the request. |
| 02H | UNSUPPORTED_VERSION | This error is returned when a request was received, whose version number is unsupported. |
| 03H | UNSUPPORTED_REQUEST | This error is returned when a request contained a Message Type that the receiver did not recognize or support. |
| 04H | POORLY_FORMED_REQUEST | This error is returned when the received message is poorly formed. The message length is not valid. |
| 05H | TIMESTAMP_MISMATCH | This error is sent when the received timestamp value is more than REPLAY_OFFSET seconds away from the receiver's local time. In this scenario, the receiver copies only the low-order 32 bits of the timestamp into the response, and supplies the high-order 32 bits from its own time of day. |
| 06H | AUTHENTICATION_FAILURE | This error is sent when message authentication failed at the receiver. |
| 07H | UNSUPPORTED_PARAMETER | The peer received a message that contained a parameter that is not recognized or supported. A message with this error MUST contain a FailedParameter element containing the IEI(s) of the one or more IEIs that caused the failure. |
| 08H | INVALID_PARAMETER_VALUE | The request contained an IE with an invalid value in its data portion. A message indicating this error MUST include the offending IEs within a FailedParameter element. |
| 09H | MISSING_PARAMETER | The request did not contain a parameter that is required by the message definition. A FailedParameter should be included in the response containing the missing IEIs. |
| 0AH | UNABLE_TO_COMPLY | This error is returned when a request is rejected for unspecified reasons. |

| 0BH | INVALID_MULTICAST_ADDR_VALUE | The AddFlowRequest message contained a value of the multicast IP address that is already assigned to another flow by the BCMCS Controller |
|---|---|---|
| 0CH | FAILURE_FOR_UNKNOWN_REASON | The Request was failure due to unknown reason. |
| **All other values are reserved.** | | |

**Table 31: ResultCode Values**

### 8.2.4.12     StartTime

This information element indicates the start time of a BCMCS Flow.  The structure of the StartTime is as shown in Table 32.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 03H | | | | | | | | 1 |
| Length = 0AH | | | | | | | | 2 |
| (MSB) | Value = < 0 - FF FF FF FF FF FF FF FFH > | | | | | | | 3 |
| | | | | | | | | ... |
| | | | | | | | (LSB) | 10 |

**Table 32: StartTime**

**IEI**: Information Element Identifier. It shall be set to 03H.

**Length**: This field indicates the number of octets in this element including IEI and Length fields. It shall be set to 0AH.

**Value**: 64-bit unsigned integer value in NTP format, as specified in RFC 1305.

### 8.2.4.13     QoSParameters

This information element indicates the QoS of BCMCS Flows.  The structure of the QoSParameters is as shown in Table 33.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 0FH | | | | | | | | 1 |
| Length = <variable> | | | | | | | | 2 |
| Identifier Type | | | | | | | | 3 |
| Identifier | | | | | | | | Variable N |
| Number of Flow Profile IDs | | | | | | | | N+1 |
| (MSB) | Flow Profile ID | | | | | | | N+2 |
| Flow Profile ID | | | | | | | | N+3 |

**Table 33: QoSParameters**

**IEI:** Information Element Identifier. It shall be set to 0FH.

**Length:** This field indicates the number of octets in this element including IEI and Length fields.

**Identifier Type:** This field shall be set to 00H if the Identifier field contains the BCMCSFlowHandler value. This filed shall be set to 04H if the Identifier field contains IPv4 Multicast Address and Port number. This filed shall be set to 06H if the Identifier field contains IPv6 Multicast Address and Port number. The sender shall set this field to 00H if the BCMCSFlowHandler is available.

**Identifier:** This field shall be set to the BCMCSFlowHandler value (4 octets) associated with the Flow Profile ID if the Identifier type field is set to 00H. This filed shall be set to IPv4 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 04H. This filed shall be set to IPv6 Multicast Address and Port number as defined in Table 19 if the Identifier field is set to 06H.

**Number of Flow Profile IDs:** This field indicates the number of Flow Profile IDs contained in this IE. This field shall be set to 1 for the CS-BCMCS Control Protocol revision 1.

**Flow Profile ID**:   One Flow Profile ID is 16 bits length. This field shall be set according to the QoS requirements for a flow. Flow Profile ID can be found in [3GPP2 C.R1001-G].

### 8.2.4.14    ContentProviderID

This information element indicates the content provider ID of BCMCS Flows.  The structure of the ContentProviderID is as shown in Table 34.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 05H | | | | | | | | 1 |
| Length | | | | | | | | 2 |
| Character Set = <00H-02H> | | | | | | | | 3 |
| (MSB) | ContentProviderID | | | | | | | 4 |
| | | | | | | | | ... |
| | | | | | | | (LSB) | N |

**Table 34: ContentProviderID**

**IEI**: Information Element Identifier. It shall be set to 05H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields.

**Character Set:** This field indicates the character set encoding type used for the Content Provider ID field as shown in Table 35.

| Value | Character Set |
|---|---|
| 00H | ASCII-8 |
| 01H | UTF-8 |
| 02H | Unicode |

**Table 35: CharacterSetValues**

**ContentProviderID:** Contains the identification (name) of the Content Provider that is the source of the content.

### 8.2.4.15    SDPParameters

This information element indicates the SDP for one or more BCMCS Flows associated with one program.  The structure of the SDPParameters is as shown in Table 36.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 0EH | | | | | | | | 1 |
| Length = <variable> | | | | | | | | 2 |
| (MSB) | SDP | | | | | | | 3 |
| ... | | | | | | | | .. |
| SDP | | | | | | | (LSB) | N |

**Table 36: SDPParameters**

**IEI:** Information Element Identifier. It shall be set to 0EH.

**Length:** This field indicates the number of octets in this element including IEI and Length fields.

**SDP:** This field shall contain the application information using the format specified in [RFC 2327].

### 8.2.4.16 LocationArea

This information element indicates the location of BCMCS Flows.  The structure of the <u>LocationArea</u> is as shown in Table 37.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 80H |  |  |  |  |  |  |  | 1 |
| Length = <variable> |  |  |  |  |  |  |  | 2 |
| P | C | S | N | PZ | | SN | CE | Resd | 3 |
| (MSB) | Country Code |  |  |  |  |  |  | 4 |
| Country Code |  |  |  |  |  |  |  | 5 |
| SID |  |  |  |  |  |  |  | 6 |
| SID |  |  |  |  |  |  |  | 7 |
| NID |  |  |  |  |  |  |  | 8 |
| NID |  |  |  |  |  |  |  | 9 |
| PZID |  |  |  |  |  |  |  | 10 |
| PZID |  |  |  |  |  |  |  | 11 |
| Subnet ID |  |  |  |  |  |  |  | 12 |
| Subnet ID |  |  |  |  |  |  |  | 13 |
| Cell ID |  |  |  |  |  |  |  | 14 |
| Cell ID |  |  |  |  |  |  | (LSB) | 15 |

**Table 37: Location Area**

**IEI:** Information Element Identifier. It shall be set to 80H.

**Length:** This field indicates the number of octets in this element including IEI and Length fields.

P (polarity): Indication of whether the associated location area is intended for positive or negative transmission.

> If Polarity = 1, this indicates the associated flow is intended for transmission within the corresponding location area.

> If Polarity = 0, this indicates the associated flow is not intended for transmission within the corresponding location area.

C: Indicates whether Country Code is present or not.

> If C = 1, this indicates the Country Code is present.

> If C = 0, this indicates the Country Code is not present.

S: Indicates whether SID is present or not.

> If S = 1, this indicates the SID is present.

> If S=0, this indicates the SID is not present.

N: Indicates whether NID is present or not.

> If N = 1, this indicates the NID is present.

> If N=0, this indicates the NID is not present.

PZ: Indicates whether PZID is present or not.

> If PZ = 1, this indicates the PZID is present.

> If PZ=0, this indicates the PZID is not present.

SN: Indicates whether Subnet ID is present or not.

> If SN = 1, this indicates the Subnet ID is present.

> If SN=0, this indicates the Subnet ID is not present.

CE: Indicates whether Cell ID is present or not.

If CE = 1, this indicates the Cell ID is present.

If CE=0, this indicates the Cell ID is not present.

**Country Code:** If C bit is set to '1', this field shall be set to the Country Code, binary coding for 1-3 digits, as specified in [OMA MLP].

**SID:** If S bit is set to '1', this field shall be set to the System ID as specified in [3GPP2 C.S0005-D].

**NID:** If N bit is set to '1', this field shall be set to the Network ID as specified in [3GPP2 C.S0005-D].

**PZID:** If PZ bot is set to '1', this field shall be set to the Packet Zone ID as specified in [3GPP2 C.S0005-D].

**Subnet ID:** If SN bit is set to '1', this field shall be set to the Subnet ID as specified in [3GPP2 C.S0024-B].

**Cell ID:** If CE bit is set to '1', this field shall be set to the Sector ID as specified in [3GPP2 C.S0024-B] or BASE_ID as specified in [3GPP2 C.S0005-D].

### 8.2.4.17    ProgramName

This information element indicates the Program Name of BCMCS Flows. The structure of the ProgramName is as shown in Table 38.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | Octet |
|---|---|---|---|---|---|---|---|---|
| Information Element Identifier (Type) = 81H | | | | | | | | 1 |
| Length | | | | | | | | 2 |
| Character Set = <00H-02H> | | | | | | | | 3 |
| (MSB) | | | | Program Name | | | | 4 |
| | | | | | | | | … |
| | | | | | | | (LSB) | N |

**Table 38: ProgramName**

**IEI**: Information Element Identifier. It shall be set to 81H.

**Length**: This field indicates the number of octets in this element including IEI and Length fields. .

**Character Set**: This field indicates the character set encoding type used for the following Program Name field, as shown in Table 39.

| Value | Character Set |
|---|---|
| 00H | ASCII-8 |
| 01H | UTF-8 |
| 02H | Unicode |

**Table 39: CharacterSetValues**

**Program Name**: Contains the name of the BCMCS program.

# 8.3    Call Flows

## 8.3.1    Add New Flow(s)

The AddFlowRequest message is sent by the BSD/A to the BCMCS Controller to request resource reservations at the BCMCS Controller for BCMCS Flow(s). The content is sent to the BCMCS CS using BSD/A-BCMCS Bearer Protocol. A successful AddFlowRequest scenario is shown in figure below.
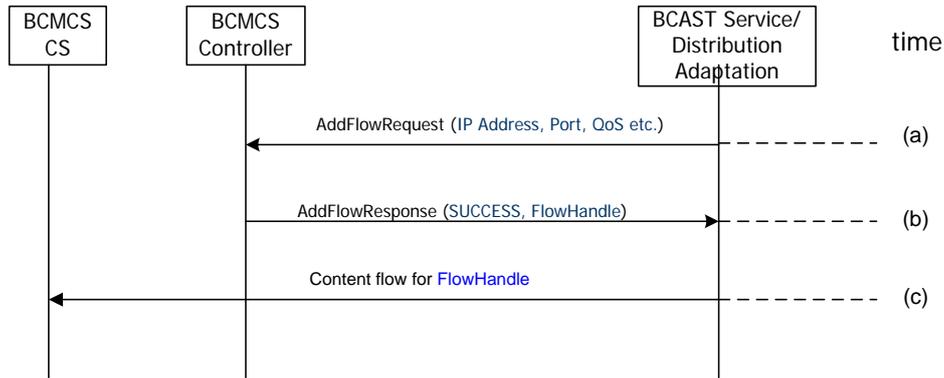


**Figure 15: Successful Add Flow Scenario**

The BSD/A is provisioned with a list of BCMCS Control Entities. When the BSD/A needs to create new BCMCS Flow(s) it uses the BSD/A-BCMCS Control Protocol.

a)  BSD/A sends an AddFlowRequest message defining parameters of the BCMCS Flow(s), such as schedule, QoS parameters, etc.
b)  BCMCS Controller sends an AddFlowResponse message accepting the BCMCS Flow reservation. The BCMCS Controller includes the BCMCS Flow Handle(s) that the BSD/A can use to reference this BCMCS Flow reservation.
c)  The BSD/A starts sending the content for the BCMCS Flow(s) at the start time of the BCMCS Flow(s) to the BCMCS CS, via the L3 Tunnel protocol.

## 8.3.2    Modify Existing Flow(s)

The ModifyFlowRequest message is sent by the BSD/A to the BCMCS Controller to modify start and end times, source IP address parameters etc. of BCMCS Flow(s) that was initially created through an AddFlowRequest message.

The use of the ModifyFlowRequest message to successfully extend the end time of a BCMCS Flow in progress is illustrated in figure below.
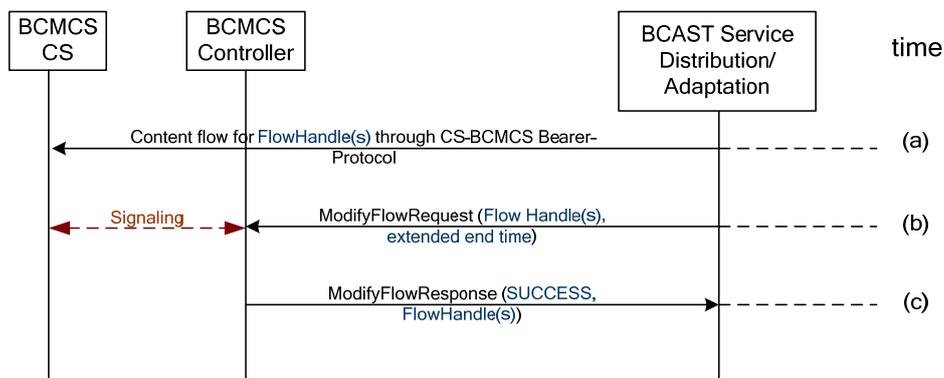


**Figure 16: Successful Modify Flow Scenario**

a) There is an existing BCMCS Flow for which the BSD/A is sending content IP datagrams to the CS.
b) The BSD/A sends a ModifyFlowRequest message to the BCMCS Controller with the new end time for the BCMCS Flow Identified by BCMCS Flow Handle.
c) The BCMCS Controller sends a ModifyFlowResponse message to indicate that the ModifyFlowRequest was successful.

The BSD/A may change parameters such as start time, end time and/or source IP address using the ModifyFlow message. However, certain changes in BCMCS Flow configuration can only be accomplished through removing a BCMCS Flow and adding a new BCMCS Flow with the new parameters.

## 8.3.3    Remove Existing Flow(s)

The RemoveFlowRequest message is sent by the BSD/A to the BCMCS Controller to remove the reservation for an existing BCMCS Flow. The successful removal of a BCMCS Flow in progress is illustrated in the figure below.
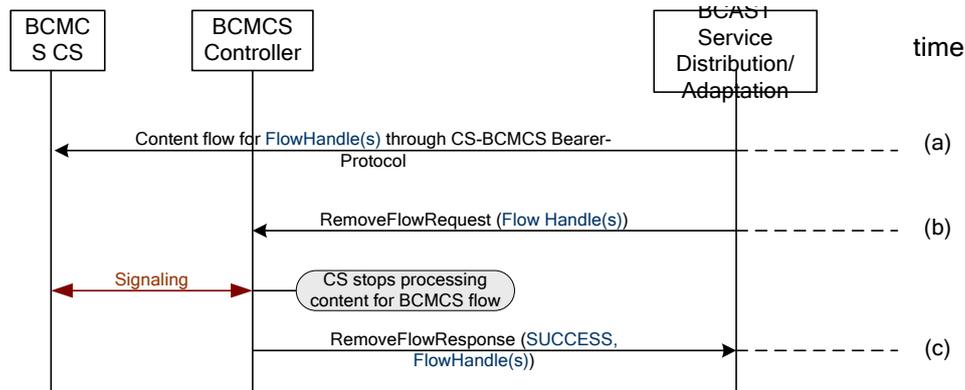


**Figure 17: Successful Remove Flow Scenario**

a) The BSD/A is sending content to the BCMCS CS on an existing BCMCS Flow Identified by a BCMCS Flow Handle.
b) The BSD/A sends a RemoveFlowRequest message that specifies the BCMCS Flow Handle to the BCMCS Controller. There is internal messaging between the BCMCS Controller and BCMCS CS that is out of scope of this document. The BCMCS CS stops processing content for the BCMCS Flow identified in RemoveFlowRequest message.
c) The BCMCS Controller sends a RemoveFlowResponse message to the BSD/A indicating successful removal of the BCMCS Flow.

# 9. Location based broadcast

Location based broadcasting in OMA BCAST are based on two types or parameters: one set relying on OMA MLP and another set relying on Cell identifier information provided by the underlying 3GPP2 network. In the case of MLP, some minor adaptation may be needed as described in this chapter.

## 9.1    Packet switched CDMA networks

The [3GPP2 X.S0024-0] defines IP-Based Location and specifies the use of OMA MLP as optional mechanism for defining the target area and shape and does not mention any other mechanism for achieving it.

If the BCMCS terminal supporting OMA BCAST Services supports [3GPP2 X.S0024-0], it SHALL support the MLP based parameters as defined in OMA Service Guide for location based broadcast.

The BCMCS terminal SHALL support the 'CellTargetArea' parameters defined in the Service and Content fragments of the Service Guide for location based broadcasting if it does not support X.S0024-0, and it MAY support the 'CellTargetArea' parameters if it supports [3GPP2 X.S0024-0].

## 9.2    Circuit switched CDMA networks

The [3GPP2 X.S0002-0] defines the Location Services Authentication/Privacy/Security and Enhancements. This specification uses shape and horizontal accuracy to define the concept of target area. The MLP derived parameters such as country code and geopolitical are replaced in the BCAST Service Guide by zip code, since [3GPP2 X.S0002-0] does not support MLP.

The BCMCS terminal SHALL support the 'CellTargetArea' parameters defined in the Service and Content fragments of the Service Guide for location based broadcasting if it does not support [3GPP2 X.S0002-0], and it MAY support the 'CellTargetArea' parameters if it supports [3GPP2 X.S0002-0].

If the BCMCS terminal supporting OMA BCAST Services supports [3GPP2 X.S0002-0] parameters, it SHALL support following Service Guide mapping scheme in the table below for location based broadcasting.

| Name | Mapped to: Name (in 3gpp2) | Type | Category | Cardinality | Description | Data Type |
|------|---------------------------|------|----------|-------------|-------------|-----------|
| Broadcast Area | broadcast_area _3gpp2 | E1 | NO/TO | 0..1 | Broadcast area to include location information for BCAST contents. Sub-elements: TargetArea Hor_acc | |
| polarity | N/A | 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 = TURE, 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 |

| Name | Mapped to: Name (in 3gpp2) | Type | Category | Cardinality | Description | Data Type |
|---|---|---|---|---|---|---|
| TargetArea | target_area_3gpp2 | E2 | NO/TM | 0..N | The target area to distribute contents (Position information as specified in the [3GPP2 X.S0002-0] with modifications) Sub-elements: Shape Cc Name_area ZipCode | |
| Shape | Shape_3gpp2 | E3 | O | 0..1 | Shapes used to represent a geographic area that describes (as specified in the [3GPP2 X.S0002-0]) | See [3GPP2 X.S0002-0] |
| Cc | Cc_BCMCS_3gpp2 | E3 | O | 0..1 | Country code, 1-3 digits e.g. 355 for Albania (as an area as specified in the [OMA MLP] and implemented as an area as defined in the [3GPP2 X.S0002-0])) | See [OMA MLP] and [3GPP2 X.S0002-0] |
| Name_area | name_area_3gpp2 | E3 | O | 0..1 | Geopolitical name of area such as 'Seoul' (as specified in the [OMA MLP] and implemented as an area as defined in the [3GPP2 X.S0002-0]) | See [OMA MLP] and [3GPP2 X.S0002-0] |
| ZipCode | zip_code_3gpp2 | E3 | O | 0..1 | Zip code (implemented as an area as defined in the [3GPP2 X.S0002-0]) | Integer |
| Hor_acc | hor_acc_3gpp2 | E2 | O | 0..N | Horizontal accuracy in meters (as specified in the [OMA MLP] and implemented as as defined in the [3GPP2 X.S0002-0]) | See [OMA MLP] and [3GPP2 X.S0002-0] |

# Appendix A.   Change History                                                  (Informative)

## A.1      Approved Version History

| Reference | Date | Description |
|---|---|---|
| OMA-TS-BCAST_BCMCS_Adaptation-V1_1-20131029-A | 29 Oct 2013 | Status changed to Approved by TP<br>TP Ref # OMA-TP-2013-0332-INP_BCAST_V1_1_ERP_for_final_Approval |

## A.2      Draft/Candidate Version 1.2 History

| Document Identifier | Date | Sections | Description |
|---|---|---|---|
| Draft Version<br>OMA-TS-BCAST_BCMCS_Adaptation-V1_2 | 10  Dec 2013 | 2.1; entire document | Version 1.2 Created from OMA-TS-BCAST_BCMCS_Adaptation-V1_1-20131029-A<br>Implementation of CR OMA-BCAST-2013-0059-CR_CONR1.2_H001_to_H005.doc |
| Candidate Version<br>OMA-TS-BCAST_BCMCS_Adaptation-V1_2 | 14 Jan 2014 | n/a | Status changed to Candidate by TP<br>TP Ref # OMA-TP-2014-0002-INP_BCAST_NGH_V1_2_ERP_and_ETR_for_Candidate_approval |

# Appendix B.   Static Conformance Requirements                (Normative)

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

Note:  The SCR entries shown below for the BCMCS Adaptation specification may overrule, or may be used to adapt to BCMCS, those requirements of SCR entries in the generic BCAST specifications.

## B.1    SCR for BCAST Terminal

Note: SCR entries referencing Section 9 of this specification are applicable to both Generic and BDS-specific adaptation modes.

| Item | Function | Reference | Status | Requirement |
|---|---|---|---|---|
| BCAST-BA-C-001 | Support for generic and BDS-specific modes of adaptation of BCAST to BCMCS | Sections 6 and 7 | O | BCAST-BAModeSig-C-002 AND BCAST-BAServices-C-003 AND BCAST-BASG-C-006 AND BCAST-BASPCPandXBS-C-010 AND BCAST-BADistrib-C-012 AND BCAST-BALoc-C-018 |
| BCAST-BAModeSig-C-002 | Signaling of the adaptation mode is performed using service provisioning procedures | Section 6 and 7 | O | |
| BCAST-BAServices-C-003 | Support for adaptation of Services TS functionality to BCMCS | Sections 6.2 and 7.2 | O | BCAST-BAServices-C-004 AND BCAST-BAServices-C-005 |
| BCAST-BAServices-C-004 | Support for adaptation of Interaction functionality to BCMCS | Sections 6.2.1 and 7.2.1 | O | BCAST-SERVICES-C-001 |
| BCAST-BAServices-C-005 | Support for adaptation of Service Provisioning functionality to BCMCS | Sections 6.2.2 and 7.2.2 | O | BCAST-SERVICES-C-006 AND BCAST-SERVICES-C-007 AND BCAST-SERVICES-C-008 |
| BCAST-BASG-C-006 | Support for adaptation of Service Guide TS functionality to BCMCS | Sections 6.3 and 7.3 | O | BCAST-BASG-C-007 AND BCAST-BASG-C-008 AND BCAST-BASG-C-009 |
| BCAST-BASG-C-007 | Support for adaptation of Service Guide delivery, over interaction channel, to BCMCS | Sections 6.3.2 and 7.3.2 | O | BCAST-SG-C-006 AND BCAST-SG-C-012 AND BCAST-SG-C-014 |
| BCAST-BASG-C-008 | Support for adaptation of Session Description Information to BCMCS | Sections 6.3.4 and 7.3.4 | O | |
| BCAST-BASG-C-009 | Support for adaptation of Service Guide Discovery to BCMCS | Sections 6.3.7 and 7.3.7 | O | |
| BCAST-BASPCPandXBS-C-010 | Support for adaptation of SPCP and XBS TS functionality to BCMCS | Sections 6.4 and 7.4 | O | BCAST-BASPCPandXBS-C-011 |
| BCAST-BASPCPandXBS-C-011 | Support for adaptation of Key Management functionality to BCMCS | Sections 6.4.2 and 7.4.2 | O | |

| Item | Function | Reference | Status | Requirement |
|---|---|---|---|---|
| BCAST-BADistrib-C-012 | Support for adaptation of File and Stream Distribution functionality to BCMCS | Sections 6.5 and 7.5 | O | BCAST-BADistrib-C-013 AND BCAST-BADistrib-C-014 AND BCAST-BADistrib-C-015 AND BCAST-BADistrib-C-017 |
| BCAST-BADistrib-C-013 | Support for BCAST-enabled Services delivered as separate BCMCS Programs | Sections 6.5.1 and 7.5.1 | O | |
| BCAST-BADistrib-C-014 | Support for adaptation of File Distribution functionality over interaction channel to BCMCS | Sections 6.5.3 and 7.5.3 | O | BCAST-FD-C-016 AND BCAST-FD-C-018 AND BCAST-FD-C-019 AND BCAST-FD-C-020 |
| BCAST-BADistrib-C-015 | Support for adaptation of Stream Distribution functionality over interaction channel to BCMCS | Sections 6.5.4 and 7.5.4 | O | BCAST-SD-C-016 AND BCAST-SD-C-020 AND BCAST-SD-C-021 |
| BCAST-BADistrib-C-016 | Support for Media CODECs and Formats | Sections 6.5.5 and 7.5.5 | O | |
| BCAST-BALoc-C-017 | Support for adaptation of Location-based Broadcast Services to BCMCS | Section 9 | O | BCAST-BALoc-C-018 AND BCAST-BALoc-C-019 OR BCAST-BALoc-C-020 OR BCAST-BALoc-C-021 |
| BCAST-BALoc-C-018 | Support for target location areas according to OMA MLP syntax if the terminal is operating in a packet-switched cdma2000 network and it supports [3GPP2 X.S0024-0] | Section 9.1 | O | |
| BCAST-BALoc-C-019 | Support for target location areas defined by 'CellTargetArea' in Service Guide, if the terminal is operating in a packet-switched cdma2000 network and it supports [3GPP2 X.S0024-0] | Section 9.1 | O | |
| BCAST-BALoc-C-020 | Support for target location areas defined by 'CellTargetArea' in Service Guide, if the terminal is operating in a circuit-switched cdma2000 network and it does not support [3GPP2 X.S0002-0] | Section 9.2 | O | |

| Item | Function | Reference | Status | Requirement |
|------|----------|-----------|--------|-------------|
| BCAST-BALoc-C-021 | Support for the Service Guide mapping scheme table of Section 9.2 for location based broadcasting if the terminal is operating in a circuit-switched cdma2000 network and it supports [3GPP2 X.S0002-0]. | Section 9.2 | O | |
| BCAST-BASpecific-C-022 | Support for BDS-specific mode of adaptation of BCAST to BCMCS | Section 7 | O | BCAST-BASPCPandXBS-C-023 AND BCAST-BASPCPandXBS-C-028 |
| BCAST-BASPCPandXBS-C-023 | Support native BCMCS service protection | Section 7.4.3.1.1 | O | BCAST-BASPCPandXBS-C-024 AND BCAST-BASPCPandXBS-C-026 |
| BCAST-BASPCPandXBS-C-024 | Support traffic encryption using SRTP for service protection,. | Section 7.4 | O | BCAST-BASPCPandXBS-C-025 |
| BCAST-BASPCPandXBS-C-025 | Support specific SRTP configuration parameters specified in Sec. 7.4.1 | Section 7.4.1 | O | |
| BCAST-BASPCPandXBS-C-026 | Support pre-provisioned key management as natively specified for BCMCS | Section 7.4.2 | O | |
| BCAST-BASPCPandXBS-C-027 | Support BCMCS Information Acquisition to obtain the SEK/PEK | Section 7.4.2.1 | O | |

# B.2 SCR for BCAST BSD/A

Note: SCR entries referencing Section 9 of this specification are applicable to both Generic and BDS-specific adaptation modes.

| Item | Function | Reference | Status | Requirement |
|------|----------|-----------|--------|-------------|
| BCAST-BABSDA-S-001 | Support for generic and BDS-specific modes of adaptation of BCAST to BCMCS | Sections 6 and 7 | O | BCAST-BABSDASG-S-002 AND BCAST-BABSDADistrib-S-006 AND BCAST-BABSDAFlowProv-S-012 AND BCAST-BABSDALoc-S-017 |
| BCAST-BABSDASG-S-002 | Support for adaptation of Service Guide TS functionality to BCMCS | Sections 6.3 and 7.3 | O | BCAST-BABSDASG-S-003 AND BCAST-BABSDASG-S-004 AND BCAST-BABSDASG-S-005 |
| BCAST-BABSDASG-S-003 | Support for adaptation of Service Guide delivery over interaction channel to BCMCS | Sections 6.3.2 and 7.3.2 | O | BCAST-SGGAD-S-002 AND BCAST-SGGAD-S-022 AND BCAST-SGGAD-S-024 |
| BCAST-BABSDASG-S-004 | Support for adaptation of Session Description Information to BCMCS | Sections 6.3.4 and 7.3.4 | O | |

| Item | Function | Reference | Status | Requirement |
|---|---|---|---|---|
| BCAST-BABSDASG-S-005 | Support for adaptation of Service Guide Discovery to BCMCS | Sections 6.3.7 and 7.3.7 | O | |
| BCAST-BABSDADistrib-S-006 | Support for adaptation of File and Stream Distribution functionality to BCMCS | Sections 6.5 and 7.5 | O | BCAST-BABSDADistrib-S-007 AND BCAST-BABSDADistrib-S-008 AND BCAST-BABSDADistrib-S-009 AND BCAST-BABSDADistrib-S-010 AND BCAST-BABSDADistrib-S-011 |
| BCAST-BABSDADistrib-S-007 | Support for BCAST-enabled Services delivered as separate BCMCS Programs | Sections 6.5.1 and 7.5.1 | O | |
| BCAST-BABSDADistrib-S-008 | Support for adaptation of File Distribution functionality over interaction channel to BCMCS | Sections 6.5.3 and 7.5.3 | O | BCAST-FD-S-025 AND BCAST-FD-S-027 AND BCAST-FD-S-028 AND BCAST-FD-S-029 |
| BCAST-BABSDADistrib-S-009 | Support for adaptation of Stream Distribution functionality over interaction channel to BCMCS | Sections 6.5.4 and 7.5.4 | O | BCAST-SD-S-025 AND BCAST-SD-S-029 AND BCAST-SD-S-030 |
| BCAST-BABSDADistrib-S-010 | Support for Media CODECs and Formats | Sections 6.5.5 and 7.5.5 | O | |
| BCAST-BABSDAFlowProv-S-011 | Support for provisioning of BCMCS Flows across BDS-1 Interface | Section 8 | O | BCAST-BABSDAFlowProv-S-012 AND BCAST-BABSDAFlowProv-S-013 AND |
| BCAST-BABSDAFlowProv-S-012 | Support BCMCS Flow Handle and associated states comprising BCMCS Flow life cycle | Section 8.1.1 | O | |
| BCAST-BABSDAFlowProv-S-013 | Support BSD/A-to-BCMCS Control Protocol, including messages and procedures triggered by bthese messages | Section 8.2 | O | BCAST-BABSDAFlowProv-S-014 AND BCAST-BABSDAFlowProv-S-015 |
| BCAST-BABSDAFlowProv-S-014 | Support BSD/A-BCMCS Control Protocol messages/message format/information elements | Section 8.2.1 8.2.3 and 8.2.4 | O | |
| BCAST-BABSDAFlowProv-S-015 | Support interaction procedures triggered by the BSD-A-BCMCS Control Protocol message exchange | Section 8.2.2 | O | |

| Item | Function | Reference | Status | Requirement |
|------|----------|-----------|--------|-------------|
| BCAST-BABSDALoc-S-016 | Support for adaptation of Location-based Broadcast Services to BCMCS | Section 9 | O | BCAST-BABSDALoc-S-017 AND BCAST-BABSDALoc-C-018 |
| BCAST-BABSDALoc-S-017 | Support for delivery in Service Guide of target location areas according to OMA MLP syntax | Section 9.1 | O | |
| BCAST-BABSDALoc-S-018 | Support for delivery in Service Guide of target location areas defined by 'CellTargetArea | Section 9.1 | O | |
| BCAST-BABSDASpecific-S-019 | Support for BDS-specific mode of adaptation of BCAST to BCMCS, with regards to Service and Content Protection | Section 7.4 | O | BCAST-BABSDASPCPandXBS-S-020 |
| BCAST-BABSDASPCPandXBS-S-020 | Support native BCMCS service protection | Section 7.4.3.1.1 | O | BCAST-BABSDASPCPandXBS-S-021 |
| BCAST-BABSDASPCPandXBS-S-021 | Support for traffic encryption using SRTP for service protection | Section 7.4 | O | BCAST-BABSDASPCPandXBS-S-022 |
| BCAST-BABSDA_SPCPandXBS-S-022 | Support specific SRTP configuration parameters specified in Sec. 7.4.1 | Section 7.4.1 | O | |

# B.3    SCR for BCAST BSM

| Item | Function | Reference | Status | Requirement |
|------|----------|-----------|--------|-------------|
| BCAST-BABSM-S-001 | Support for generic and BDS-specific modes of adaptation of BCAST to BCMCS | Sections 6 and 7 | O | BCAST-BABSMModeSig-S-002 AND BCAST-BABSMServices-S-003 |
| BCAST-BABSMModeSig-S-002 | Signaling of the adaptation mode is performed using service provisioning procedures | Section 6 and 7 | O | |
| BCAST-BABSMServices-S-003 | Support subscription to Service Guide and Service Guide Delivery Descriptor as BCMCS Programs | Sections 6.3.7 and 7.3.7 | O | |
| BCAST-BABSMSPCP&XBS-S-004 | Support "Subscriber Key Establishment using a (R-)UIM/CSIM" key establishment procedures | Sections 6.4 and 7.4 | O | |
| BCAST-BABSMSpecific-S-005 | Support for BDS-specific mode of adaptation of BCAST to BCMCS, with regards to Service and Content Protection | Section 7.4.2 | O | BCAST-BABSMSPCP&XBS-S-006 |

| Item | Function | Reference | Status | Requirement |
|---|---|---|---|---|
| BCAST-BABSMSPCP&XBS-S-006 | Support "Subscriber Key Establishment using a (R-)UIM/CSIM" key establishment procedures, based on pre-provisioned key management as specified in native BCMCS | Section 7.4.2 | O | |