3GPP TR 23.792 V16.0.0 (2018-06)

Technical Report

3rd Generation Partnership Project;

Technical Specification Group Services and System Aspects;

Study on MBMS APIs for Mission Critical Services;

(Release 16)

** 

The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP.  
The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented.  
This Report is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification.  
Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices.

Keywords

<keyword[, keyword]>

***3GPP***

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis

Valbonne - FRANCE

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

Internet

http://www.3gpp.org

***Copyright Notification***

No part may be reproduced except as authorized by written permission.  
The copyright and the foregoing restriction extend to reproduction in all media.

© 2018, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).

All rights reserved.

UMTS™ is a Trade Mark of ETSI registered for the benefit of its members

3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  
LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners

GSM® and the GSM logo are registered and owned by the GSM Association

Contents

Foreword [6](#__RefHeading___Toc517445550)

Introduction [6](#__RefHeading___Toc517445551)

1 Scope [7](#__RefHeading___Toc517445552)

2 References [7](#__RefHeading___Toc517445553)

3 Definitions and abbreviations [8](#__RefHeading___Toc517445554)

3.1 Definitions [8](#__RefHeading___Toc517445555)

3.2 Abbreviations [8](#__RefHeading___Toc517445556)

4 Requirements and reference model [8](#__RefHeading___Toc517445557)

4.1 Functional requirements [8](#__RefHeading___Toc517445558)

4.2 Reference model [9](#__RefHeading___Toc517445559)

4.2.1 General [9](#__RefHeading___Toc517445560)

4.2.2 MC application [9](#__RefHeading___Toc517445561)

4.2.3 MC MBMS user agent [10](#__RefHeading___Toc517445562)

4.3 Deployment considerations [10](#__RefHeading___Toc517445563)

5 Key issues and gaps [10](#__RefHeading___Toc517445564)

5.1 Announcement handling [10](#__RefHeading___Toc517445565)

5.2 ROHC and FEC handling [10](#__RefHeading___Toc517445566)

5.3 Feedback on MBMS reception [10](#__RefHeading___Toc517445567)

5.4 API for file distribution over MBMS [11](#__RefHeading___Toc517445568)

5.5 Registration [11](#__RefHeading___Toc517445569)

5.6 MBMS bearer quality monitoring [11](#__RefHeading___Toc517445570)

5.7 Location information [11](#__RefHeading___Toc517445571)

6 Use cases [11](#__RefHeading___Toc517445572)

6.0 General [11](#__RefHeading___Toc517445573)

6.1 MBMS bearer quality monitoring and data reception [12](#__RefHeading___Toc517445574)

6.1.1 General description [12](#__RefHeading___Toc517445575)

6.1.2 Typical flows [12](#__RefHeading___Toc517445576)

6.2 Location information [13](#__RefHeading___Toc517445577)

6.2.1 General description [13](#__RefHeading___Toc517445578)

6.2.2 Typical flows [13](#__RefHeading___Toc517445579)

6.3 Switching from MBMS delivery to unicast delivery [13](#__RefHeading___Toc517445580)

6.3.1 General description [13](#__RefHeading___Toc517445581)

6.3.2 Typical flows [14](#__RefHeading___Toc517445582)

6.4 MBMS bearer suspension and resumption [14](#__RefHeading___Toc517445583)

6.4.1 General description [14](#__RefHeading___Toc517445584)

6.4.2 Typical flows [14](#__RefHeading___Toc517445585)

6.5 Using the MBMS user services for MCData file distribution [15](#__RefHeading___Toc517445586)

6.5.1 General [15](#__RefHeading___Toc517445587)

6.5.2 Architecture [15](#__RefHeading___Toc517445588)

6.5.3 Typical flows [16](#__RefHeading___Toc517445589)

6.5.3.1 Group standalone file distribution using MBMS download delivery method. [16](#__RefHeading___Toc517445590)

7 Solutions [18](#__RefHeading___Toc517445591)

7.1 Registration and deregistration API [18](#__RefHeading___Toc517445592)

7.1.1 General [18](#__RefHeading___Toc517445593)

7.1.2 Procedures [18](#__RefHeading___Toc517445594)

7.1.2.1 Application registration [18](#__RefHeading___Toc517445595)

7.1.2.2 Application deregistration [19](#__RefHeading___Toc517445596)

7.1.3 Information flows [19](#__RefHeading___Toc517445597)

7.1.3.1 Application registration request [19](#__RefHeading___Toc517445598)

7.1.3.2 Application registration response [19](#__RefHeading___Toc517445599)

7.1.3.3 Application deregistration request [20](#__RefHeading___Toc517445600)

7.1.4 Evaluation [20](#__RefHeading___Toc517445601)

7.2 MBMS bearer registration and notification API [20](#__RefHeading___Toc517445602)

7.2.1 General [20](#__RefHeading___Toc517445603)

7.2.2 Procedures [20](#__RefHeading___Toc517445604)

7.2.2.1 MC MBMS bearer registration [20](#__RefHeading___Toc517445605)

7.2.2.2 MC MBMS bearer deregistration [21](#__RefHeading___Toc517445606)

7.2.2.3 MC MBMS bearer availability [22](#__RefHeading___Toc517445607)

7.2.2.4 MC MBMS bearer unavailability [23](#__RefHeading___Toc517445608)

7.2.2.5 MC MBMS bearer quality evaluation [23](#__RefHeading___Toc517445609)

7.2.3 Information flows [24](#__RefHeading___Toc517445610)

7.2.3.1 MC MBMS bearer registration request [24](#__RefHeading___Toc517445611)

7.2.3.2 MC MBMS bearer registration response [24](#__RefHeading___Toc517445612)

7.2.3.3 MC MBMS bearer deregistration request [25](#__RefHeading___Toc517445613)

7.2.3.4 MC MBMS bearer deregistration response [25](#__RefHeading___Toc517445614)

7.2.3.5 MC MBMS bearer event notification [25](#__RefHeading___Toc517445615)

7.2.4 Evaluation [25](#__RefHeading___Toc517445616)

7.3 API for media reception [25](#__RefHeading___Toc517445617)

7.3.1 General [25](#__RefHeading___Toc517445618)

7.3.2 Procedures [26](#__RefHeading___Toc517445619)

7.3.2.1 Open media [26](#__RefHeading___Toc517445620)

7.3.2.2 Close media [27](#__RefHeading___Toc517445621)

7.3.3 Information flows [27](#__RefHeading___Toc517445622)

7.3.3.1 Open media request [27](#__RefHeading___Toc517445623)

7.3.3.2 Open media response [28](#__RefHeading___Toc517445624)

7.3.3.3 Close media request [28](#__RefHeading___Toc517445625)

7.3.3.4 Close media response [28](#__RefHeading___Toc517445626)

7.3.4 Evaluation [28](#__RefHeading___Toc517445627)

7.4 Location information API [29](#__RefHeading___Toc517445628)

7.4.1 General [29](#__RefHeading___Toc517445629)

7.4.2 Procedures [29](#__RefHeading___Toc517445630)

7.4.2.1 Get SAI [29](#__RefHeading___Toc517445631)

7.4.2.2 SAI update notification [29](#__RefHeading___Toc517445632)

7.4.2.3 Get cell ID [30](#__RefHeading___Toc517445633)

7.4.2.4 Cell ID update notification [30](#__RefHeading___Toc517445634)

7.4.3 Information flows [31](#__RefHeading___Toc517445635)

7.4.3.1 Get SAI response [31](#__RefHeading___Toc517445636)

7.4.3.2 SAI update notification [31](#__RefHeading___Toc517445637)

7.4.3.3 Get cell ID response [31](#__RefHeading___Toc517445638)

7.4.3.4 Cell ID update notification [32](#__RefHeading___Toc517445639)

7.4.4 Evaluation [32](#__RefHeading___Toc517445640)

7.5 MBMS API for MCData file distribution [32](#__RefHeading___Toc517445641)

7.5.1 General [32](#__RefHeading___Toc517445642)

7.5.2 Application registration and deregistration [32](#__RefHeading___Toc517445643)

7.5.3 File reception [33](#__RefHeading___Toc517445644)

7.5.4 Evaluation [33](#__RefHeading___Toc517445645)

8 Evaluations [33](#__RefHeading___Toc517445646)

8.1 API solution evaluation [33](#__RefHeading___Toc517445647)

8.2 Review of existing MBMS API and capabilities [34](#__RefHeading___Toc517445648)

9 Conclusions [34](#__RefHeading___Toc517445649)

Annex A: Analysis of capabilities provided by the MBMS user service architecture [34](#__RefHeading___Toc517445650)

A.1 General [34](#__RefHeading___Toc517445651)

A.2 Announcement / User service discovery [35](#__RefHeading___Toc517445652)

A.3 MBMS delivery methods [35](#__RefHeading___Toc517445653)

A.3.1 MBMS streaming delivery method [35](#__RefHeading___Toc517445654)

A.3.2 MBMS download delivery method [36](#__RefHeading___Toc517445655)

A.3.3 Transparent mode (transport only) [36](#__RefHeading___Toc517445656)

A.4 xMB API [37](#__RefHeading___Toc517445657)

A.5 MBMS API [37](#__RefHeading___Toc517445658)

A.6 GCSE architecture [38](#__RefHeading___Toc517445659)

A.7 Gap analysis [38](#__RefHeading___Toc517445660)

A.7.1 xMB [38](#__RefHeading___Toc517445661)

A.7.1.1 Usage of pre-established bearer [38](#__RefHeading___Toc517445662)

A.7.1.2 Geographical area [38](#__RefHeading___Toc517445663)

A.7.1.3 QoS [38](#__RefHeading___Toc517445664)

A.7.1.4 Header compression [38](#__RefHeading___Toc517445665)

A.7.1.5 FEC [38](#__RefHeading___Toc517445666)

A.7.1.6 MCData file distribution [38](#__RefHeading___Toc517445667)

A.7.2 MBMS API [39](#__RefHeading___Toc517445668)

A.7.2.1 Service identity [39](#__RefHeading___Toc517445669)

A.7.2.2 MBMS quality detection [39](#__RefHeading___Toc517445670)

A.7.2.3 Header decompression [39](#__RefHeading___Toc517445671)

A.7.2.4 FEC [39](#__RefHeading___Toc517445672)

A.7.2.5 Announcement [39](#__RefHeading___Toc517445673)

A.7.2.6 Location reporting [39](#__RefHeading___Toc517445674)

Annex B: Change history [40](#__RefHeading___Toc517445675)

# Foreword

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

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

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

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

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

# Introduction

This clause is optional. If it exists, it is always the second unnumbered clause.

# 1 Scope

The present document is a technical report that identifies the MBMS API solutions on the UE for mission critical applications to access MBMS bearer service for the usage of mission critical services. The aspects of the study include identifying functional requirements, methods and message flows, client reference model, potential impacts to MC system architecture, and the review of existing and new MBMS capabilities for use by MC services.

# 2 References

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

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

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

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

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

[2] 3GPP TS 22.282: "Mission Critical Data services".

[3] 3GPP TS 23.280: "Common functional architecture to support mission critical services; Stage 2".

[4] 3GPP TS 23.281: "Functional architecture and information flows to support Mission Critical Video (MCVideo); Stage 2".

[5] 3GPP TS 23.282: "Functional architecture and information flows to support Mission Critical Data (MCData); Stage 2".

[6] 3GPP TS 23.379: "Functional architecture and information flows to support Mission Critical Push To Talk (MCPTT); Stage 2".

[7] 3GPP TS 23.468: "Group Communication System Enablers for LTE (GCSE\_LTE); Stage 2".

[8] 3GPP TS 26.346: "Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs".

[9] 3GPP TS 26.347: "Multimedia Broadcast/Multicast Service (MBMS); Application Programming Interface and URL".

[10] 3GPP TS 29.116: "Representational state transfer over xMB reference point between content provider and BM-SC".

[11] 3GPP TS 29.468: "Group Communication System Enablers for LTE (GCSE\_LTE); MB2 reference point; Stage 3".

[12] 3GPP TS 36.300: "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2".

[13] IETF RFC 3926: "FLUTE - File Delivery over Unidirectional Transport".

[14] IETF RFC 6363: "Forward Error Correction (FEC) Framework".

[15] IETF RFC 6681: "Raptor FEC Schemes for FECFRAME".

# 3 Definitions and abbreviations

## 3.1 Definitions

**MC MBMS API:** An application programming interface on the UE to enable the use of MBMS bearer service for mission critical applications.

**MBMS bearer availability:** Presence of the TMGI identifying an MBMS bearer within the radio control channels.

**MBMS bearer quality monitoring:** Continuous evaluation of the reception quality of an MBMS bearer, with or without data being received on the MBMS bearer.

## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

API Application Programming Interface

BM-SC Broadcast Multicast Service Centre

ECGI E-UTRAN Cell Global Identifier

FEC Forward Error Correction

FLUTE File deLivery over Unidirectional Transport

MBMS Multimedia Broadcast and Multicast Service

MCCH Multicast Control Channel

MCData Mission Critical Data

MCPTT Mission Critical Push To Talk

MCVideo Mission Critical Video

ROHC Robust Header Compression

SACH Service Announcement Channel

SAI Service Area Identity

SC-MCCH Single Cell Multicast Control Channel

SDP Session Description Protocol

TMGI Temporary Mobile Group Identity

USD User Service Description

# 4 Requirements and reference model

## 4.1 Functional requirements

[MBMSAPI-FR-1] The MC MBMS API shall support registration and de-registration of MC service applications, for the use of MBMS services on the UE.

[MBMSAPI-FR-2] The MC MBMS API shall provide mechanisms to notify the MC applications the multicast location information of the MC service UE.

[MBMSAPI-FR-3] The MC MBMS API shall provides mechanisms for the MC applications to query the multicast location information of the MC service UE.

NOTE: The multicast location information can be MBMS SAI, ECGI or other information that is decoded from the broadcast information of the local cell (e.g. SIB1 and SIB15) of the MC service UE.

[MBMSAPI-FR-4] The MC MBMS API shall provide mechanisms to notify the MC applications about the MBMS bearer announcement and de-announcement that is sent over MBMS.

[MBMSAPI-FR-5] The MC MBMS API shall support mechanisms for the MC applications to start and stop monitoring the MBMS bearer quality.

[MBMSAPI-FR-6] The MC MBMS API shall be asynchronous for the MC applications.

[MBMSAPI-FR-7] The MC MBMS API shall support event notification of MBMS bearer quality detection.

[MBMSAPI-FR-8] The MC MBMS API shall support event notification of MBMS suspension and MBMS resumption.

[MBMSAPI-FR-9] The MC MBMS API shall support mechanisms for the MC applications to receive the MBMS data from the MC MBMS user agent.

[MBMSAPI-FR-10] The MC MBMS user agent shall support multiple MC applications to use the MC MBMS API.

[MBMSAPI-FR-11] The MC MBMS API shall allow concurrent calls without blocking the calling application.

## 4.2 Reference model

### 4.2.1 General



Figure 4.2.1-1: MC MBMS API reference model

In the reference model the MC application can represent an MC service client.

The MC MBMS user agent supports the MC service client functions in receiving the MBMS bearer data for the use of MBMS transmission defined in 3GPP TS 23.280 [3], 3GPP TS 23.379 [6], 3GPP TS 23.281 [4] and 3GPP TS 23.282 [5].

### 4.2.2 MC application

The MC application invokes the MC MBMS API to access and manage the MBMS bearers to support MC services.

The MC application supports the following functions:

- Registration and deregistration to the MC MBMS user agent;

- Reception of application level control signalling messages;

- Activation and termination of MBMS bearer quality monitoring;

- Obtaining MBMS bearer availability through the MC MBMS API;

- Obtaining MBMS bearer suspension and resumption status through the MC MBMS API;

- Reception of media with or without FEC encoding and media with or without ROHC compression from the MC MBMS user agent;

- Obtaining location information for MBMS usage through the MC MBMS API; and

- Configuration of the MC MBMS user agent.

### 4.2.3 MC MBMS user agent

The MC MBMS user agent may provide MBMS media with FEC and ROHC to the MC application.

One or more MC application(s) use the MC MBMS API provided by the MC MBMS user agent.

The MC MBMS user agent supports the following functions:

- Detection of location information for MBMS usage;

- Detection of MBMS bearer availability;

- Report of MBMS bearer quality to the MC application;

- Reception of MBMS data from a lower layer of the UE;

- Detection of suspension of MBMS bearer; and

- Delivery of media with or without FEC encoding and media with or without ROHC compression to the MC application.

## 4.3 Deployment considerations

No deployment issues have been identified in the present document.

# 5 Key issues and gaps

## 5.1 Announcement handling

The announcement of MBMS bearers for mission critical services may be done on a unicast bearer and/or on an MBMS bearer (3GPP TS 23.280 [3]). It should be studied how the MC MBMS API can be made to support the reception of the announcement and de-announcement over MBMS.

## 5.2 ROHC and FEC handling

When ROHC is applied on a media delivered over MBMS, header decompression is currently performed by the MC service client (see 3GPP TS 23.280 [3], subclause 10.7.3.12).

When application layer FEC is used as protection against the packet losses, the MC service client is currently in charge of FEC decoding.

The FEC and ROHC capability can be supported by either the MC application, or the MC MBMS user agent, or can be supported by both.

Both header decompression and FEC decoding are specific functions to MBMS delivery, to optimize of the MBMS channel usage. It should be studied how the MC MBMS API can be made to support these functions.

## 5.3 Feedback on MBMS reception

3GPP TS 23.280 [3] introduces three procedures where the MC service client provides feedback related to the availability and quality of MBMS bearers:

a) MBMS bearer quality detection (subclause10.7.3.6);

b) MBMS suspension notification (subclause 10.7.3.8); and

c) switching from MBMS bearer to unicast bearer (subclause 10.7.3.3).

These procedures rely on a MC service client ability to detect availability of the MBMS bearer and evaluate the reception quality. It should be studied how the MC MBMS API can be made to support this ability.

## 5.4 API for file distribution over MBMS

MCData file distribution capability can deliver files over MBMS through the MCData-FD-3 reference point with the procedure "group standalone file distribution using media plane" (3GPP TS 23.282 [5]).

Implementing a file distribution capability over a unidirectional channel such as MBMS covers:

1) An unidirectional file delivery protocol.

2) A file repair procedure, to achieve efficiently the service reliability.

3) A reception report procedure, to acknowledge the reception, as required in [R-5.3.2-006] (3GPP TS 22.282 [2]).

4) Possibly application layer FEC.

5) Possibly the announcement of the delivery schedule.

Each of these points is specific to the delivery over MBMS. It should be studied how the MC MBMS API can be made to support them.

## 5.5 Registration

Registration and de-registration is required to perform initialization and termination operations for using the MC MBMS API. The purpose of registration is to allow access of MBMS service on the UE for MC applications. This registration also uniquely identifies each MC application that can access MBMS service on the UE. De-registration is used to stop access of MBMS service on the UE for MC applications.

Further study is required to address the above key issue.

## 5.6 MBMS bearer quality monitoring

There may be time span between an MBMS bearer to be announced and mapped with a group. The UE should be able to monitor the MBMS bearer quality with or without data being received on the MBMS bearer. Further study is needed to address the issue.

## 5.7 Location information

For MBMS usage, the MC service server needs to know the location (e.g. SAI, ECGI) of the UE. Such location information is acquired by the MC service UE and reported to the MC service client.

It should be studied how the MC MBMS API can support the delivery of the location information to the MC application.

# 6 Use cases

## 6.0 General

The MC application shown in the diagrams in the use cases is functioning as a MC service client.

## 6.1 MBMS bearer quality monitoring and data reception

### 6.1.1 General description

This use case depicts the API interactions that can be applied with MBMS usage of MBMS bearer announcement, MBMS bearer establishment and application level control signalling over MBMS defined in 3GPP TS 23.280 [3]. Relevant key issues includes "Announcement handling", "ROHC and FEC handling", "MBMS bearer quality monitoring" and "Feedback on MBMS reception".

### 6.1.2 Typical flows

Figure 6.1.2-1 demonstrates the typical flow for the use case of MBMS monitoring and data reception.



Figure 6.1.2-1: Typical flow for MBMS bearer quality monitoring and data reception

1. The MBMS bearer(s) has been activated and announced by the MC service server as defined in 3GPP TS 23.280 [3].

2. API call is triggered to the MC MBMS user agent to check MBMS bearer availability and to initiate MBMS bearer quality monitoring.

3. The MC service client may send MBMS listening status report to the MC service server.

4. API call is triggered to the MC MBMS user agent to start reception of application level control signalling over MBMS bearer.

5. The MC service client may send MBMS listening status report to the MC service server.

6. The MC service server sends MapGroupToBearer message to the MC service client.

7. API call is triggered to the MC MBMS user agent to start reception of MC media over an MBMS bearer.

NOTE: The API calls in step 4 and step 7 could be of the same type, but with parameters to request different user plane media of the MBMS bearer.

8. The MC service client may send MBMS listening status report to the MC service server.

9. The MC application begin receiving multicast media from the MC MBMS client through the MC MBMS API.

Editor's note: Parameters carried in the API messages are FFS.

## 6.2 Location information

### 6.2.1 General description

This use case depicts the API interactions of MC application obtaining the location information for MBMS usage. This use case can be applied with MBMS bearer establishement (including pre-established and dynamic MBMS bearer establishment) defined in 3GPP TS 23.280 [3].

### 6.2.2 Typical flows

Figure 6.2.2-1 demonstrates the typical flow for the use case of location information. In this typical flow the UE's location information is obtained by the MC application though the API call, and reported to the MC service server by mechanisms defined in 3GPP TS 23.280 [3]. The MC service server may make use of this location information to make decisions for, for example MBMS bearer establishment, or switching between MBMS delivery and unicast delivery.

Pre-conditions:

- The MC application has registered to the MC MBMS user agent.



Figure 6.2.2-1: Typical flow for location information

1. The MC application obtains the location information of the MC service UE through the MC MBMS API.

2. The location information is reported to the MC service server based on mechanisms defined in 3GPP TS 23.280 [3].

NOTE: For simplicity of presentation, the entities relevant to location management are not shown in this flow diagram.

## 6.3 Switching from MBMS delivery to unicast delivery

### 6.3.1 General description

This use case depicts the API interactions that can be applied with MBMS usage of switching from MBMS delivery to unicast delivery defined in 3GPP TS 23.280 [3]. Relevant key issues includes "MBMS bearer quality monitoring" and "Feedback on MBMS reception".

### 6.3.2 Typical flows

Figure 6.3.2-1 demonstrates the typical flow for the use case of switching from MBMS delivery to unicast delivery.

Pre-conditions:

- The MC MBMS user agent is monitoring the MBMS bearer quality.

- The MC application is involved in the group call and is receiving media over MBMS.



Figure 6.3.2-1: Typical flow for switching from MBMS delivery to unicast delivery

1. The MC application detects bad MBMS bearer condition through the MC MBMS API.

2. The MC application notifies the MC service server about the MBMS bearer condition by sending the MBMS listening status report.

3. The MC service server decides to send the downlink data by unicast delivery instead of MBMS delivery to the MC service client.

4. The MC application may stop receiving downlink data over MBMS through the MC MBMS API interaction.

## 6.4 MBMS bearer suspension and resumption

### 6.4.1 General description

This use case depicts the API interactions that can be applied by using MBMS bearer suspension and resumption as defined in 3GPP TS 23.280 [3]. The relevant key issue is addressed in subclause 5.3 Feedback on MBMS reception.

### 6.4.2 Typical flows

Figure 6.4.2-1 demonstrates the typical flow for the use case of MBMS bearer suspension and resumption.

Pre-conditions:

- The reporting for suspension or resumption of MBMS bearer is enabled for the MC service client.



Figure 6.4.2-1: Typical flow for MBMS bearer suspension and resumption

1. The MC MBMS user agent detects suspension or resumption of an MBMS bearer.

NOTE: Indication of MBMS suspension on the UE is defined in 3GPP TS 36.300 [12].

2. The MC MBMS user agent indicates the MC application about MBMS bearer suspension or resumption via the MC MBMS API.

3. The MC application sends an MBMS suspension report to the MC service server.

## 6.5 Using the MBMS user services for MCData file distribution

### 6.5.1 General

This use case depicts the reuse of the existing file delivery mecanism named MBMS download delivery method, thanks to the MBMS user service architecture.

It does not require any new API within the MC MBMS user agent, but reuse the MBMS API.

### 6.5.2 Architecture

This use case makes use of the end-to-end Architecture for Application Service Providers using eMBMS for Delivery.



Figure 6.5.2-1: End-to-end Architecture for Application Service Providers using eMBMS for Delivery (3GPP TS 26.347 [9])

This architecture is also named MBMS user service architecture.

In this use case, the MCData server for file distribution is a content provider/application service provider as defined in 3GPP TS 26.346 [8] and 3GPP TS 26.347 [9]. The MC application is an MBMS aware application, making use of the MBMS API to receive the distributed files. The solution only makes use of the existing MBMS API and does not propose any new API exposed by the MC MBMS user agent.

This use case requires the addition of the MBMS user service architecture (xMB, MBMS API reference points) within the existing on-network architecture as shown in figure 6.5.2-2.



Figure 6.5.2-2: MCData on-network architecture including the MBMS user service architecture.

The use case respects the security endpoints, if delivered files are encrypted before being provided to the BM-SC, and only encrypted files are cached within the MBMS client.

The use case requires that the geographical areas provided by the MCData AS to the BM-SC, whose content is left to a business agreement between the content provider and the operator (see gap A.7.1.2), allows the MCData AS to control exactly the coverage of its MBMS bearer.

The use case requires also that the QoS parameters agreed between the content provider (MCData AS) and the operator (BM-SC) matches the QoS requirements for MCData file distribution (see gap A.7.1.3).

### 6.5.3 Typical flows

#### 6.5.3.1 Group standalone file distribution using MBMS download delivery method.

This flow demonstrates the interactions between the MCData server, the BM-SC, the MBMS client and the MC application to deliver a file to the affiliated members of a group, using the MBMS download delivery method.

Pre-conditions:

1. The MCData users on the MCData client 1 to n belong to the same group and are already registered for receiving MCData service and affiliated.

2. The file to be distributed, originated by a group member, is uploaded to the media storage function on MCData server.

3. The MCData server has decided to deliver the file to the group members is with the MBMS download delivery method. This MCData server decision can be based on the member locations, the number of target UEs within a given area.



Figure 6.5.3.1-1: Group standalone FD using MBMS download delivery method

1. The MCData server creates a MBMS user service over xMB, providing its identifier: the MBMS user service id (see subclause 5.4A.4.2 in 3GPP TS 26.346 [8]), and a session for this MBMS user service (see subclause 5.4A.4.2 in 3GPP TS 26.346 [8]), with Session Type set to "Files" and ingest-mode set to "Push" (see subclause 5.4A.4.2 in 3GPP TS 26.346 [8]). The create session request also contains the list of geographical areas where to deliver over MBMS and the allocated bit rate. In response the BM-SC communicates an URL where to the MCData server can push to file to be delivered over this MBMS user service.

NOTE : If such an MBMS user service with its session is already established for the given list geographical areas, the MCData can decide to deliver over this MBMS service without creating a new one.

2. The BM-SC updates the MBMS user service description (USD) delivered of the service announcement channel (SACH), with the description of the MBMS user service, including the service id, the TMGI and other delivery parameters defined in 3GPP TS 26.346 [8] and creates the MBMS bearer.

3. MCData server notifies each MCData user affiliated to this group about the file delivery. The notification contains the MBMS user service id used for the delivery and the file information.

NOTE: These signaling messages could be sent over unicast, or over an MBMS bearer for application control signalling as defined in 3GPP TS 23.280 [3].

4. The MCData client registers towards the MBMS client, if not already done, with the registerFdApp method (see 3GPP TS 26.347 [9]) and calls the startFdCapture methods, providing the MBMS user service id and the file URL.

5. The MBMS client gets the USD provided by the SACH for this MBMS user service id and initiates the file capture according 3GPP TS 26.346 [8].

6. The MCData server pushes the file to the BM-SC.

7. The BM-SC delivers the file over the MBMS service with the MBMS download delivery method. The MBMS client acquires the file.

8. When fully received, the MBMS client notifies the MCData client about the file availability with the fileAvailable notification (see 3GPP TS 26.347 [9]). The MCData clients record file download completed and notify the MCData users.

9. The MCData client(s) send file download completed report, if requested.

# 7 Solutions

## 7.1 Registration and deregistration API

### 7.1.1 General

This solution covers key issue 5.5 – Registration. The purpose of application registration is, in particular, to provide to the MC MBMS user agent the call back listeners to receive asynchronous notifications on relevant events.

### 7.1.2 Procedures

#### 7.1.2.1 Application registration

This procedure allows the MC application to register with the MC MBMS user agent to consume MC services delivers over MBMS. It has two purposes:

1. It identifies the MC application registering with the MC MBMS user agent.

2. It allows the MC application to identify its callback listeners for the MC MBMS user agent to provide asynchronous notifications on relevant events.

NOTE: Since some application development frameworks do not support callback functions, an MC application for these frameworks will not provide callback listeners in this procedure. Instead, the MC application will implement the necessary approach available on these frameworks to receive event notifications from the MC MBMS user agent in place of callback functions. The notifications implemented on these frameworks will include the same information content.

Pre-conditions:

1. The MC application has discovered the availability of the MC MBMS API.



Figure 7.1.2.1-1: Application registration procedure

1. MC application registers to the MC MBMS user agent with the application registration request.

2. MC MBMS user agent registers the application

3. Registration result is notified to the MC application.

Post-conditions:

1. If registration is successful, the MC application can proceed with other MC MBMS API interactions with the MC MBMS user agent.

#### 7.1.2.2 Application deregistration

API usage is illustrated in the registration / deregistration procedure:

Pre-conditions:

1. The MC application is registered towards the MC MBMS user agent.



Figure 7.1.2.2-1: Application deregistration procedure

1. The MC application deregisters. No additional notification from the MC MBMS user agent is expected by the MC application.

2. The MC MBMS user agent interrupts all operations previously requested by the MC application (e.g. MBMS bearer monitoring or consumption) and deletes any parameters and/or internal state associated to this application.

Post-conditions:

1. The MC application and can no more proceed any API interactions with the MC MBMS user agent till a new execution of the application registration procedure.

### 7.1.3 Information flows

#### 7.1.3.1 Application registration request

Table 7.1.3.1-1 describes the information flow for the application registration request.

Table 7.1.3.1-1: Application registration request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Application ID | M | Provides a unique ID for the application registering with the MC MBMS user agent. |
| Group communication callback | O (NOTE) | Provides the call back listener. |
| Application specific context | O | Enables the MC MBMS user agent to distinguish different applications. |
| NOTE: The callback element is optional and only included when the application development framework supports programmatic callback interfaces. If callbacks are not supported on a given application development framework, the same information content as defined on the callback structures is to be provided to the application via the notification callback available with that development framework when the respective condition is met. | | |

Editor's note: The use of application specific context is FFS.

Editor's note: Need of additional IEs for configuration is FFS.

#### 7.1.3.2 Application registration response

Table 7.1.3.2-1 describes the information flow for the application registration response.

Table 7.1.3.2-1: Application registration response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Response code | M | Provides the result of the registration request: success, failure or missing parameters. |
| Message | O | Provides an associated text description of the error message, if registration failed |

#### 7.1.3.3 Application deregistration request

There are no information elements within the application deregistration request.

### 7.1.4 Evaluation

This solution introduces a simple and non blocking API for registration and deregistration. It follows the general principles of the MBMS-API: registration of the application, usage of callbacks for notifications if possible with the application development framework.

## 7.2 MBMS bearer registration and notification API

### 7.2.1 General

This solution covers key issue 5.3 about feedback on MBMS reception.

It introduces a set of procedures within the MC MBMS API, to exchange information about the announced MBMS bearers for which availability has to be checked and quality has to be monitored. It also introduced an associated set of notification procedures.

This solution assumes that a notification mechanism has been initialized during the application registration (key issue 5.5).

### 7.2.2 Procedures

#### 7.2.2.1 MC MBMS bearer registration

This procedure allows the MC application to inform the MC MBMS user agent about an MBMS bearer announced to the MC application.

Pre-conditions:

1. The MC application is registered towards the MC MBMS user agent.

2. An MC MBMS bearer has been announced to the MC application by the MC service server.



Figure 7.2.2.1-1: MC MBMS bearer registration procedure

1. The MC application registers a newly announced MBMS bearer to the MC MBMS user agent. The MC application informs the MC MBMS user agent if reception quality evaluation is expected from the MC MBMS user agent.

2. The MC MBMS user agent registers the new MBMS bearer.

Note: If the MBMS bearer is announced with a list of alternative TMGIs, the MC application can register the MBMS bearer several times, for each distinct TMGI.

3. The MC MBMS user agent notifies the MC application about the registration result.

Post-conditions:

1. If registration is successful, the MC MBMS user agent checks the MBMS bearer presence and execute MC MBMS bearer availability procedure or MC MBMS bearer unavailability procedure.

#### 7.2.2.2 MC MBMS bearer deregistration

This procedure allows the MC application to deregister a MBMS bearer with the MC MBMS user agent. The MC application can execute this procedure when this MBMS bearer has been de-announced.

Pre-conditions:

1. The MC application is registered towards the MC MBMS user agent.

2. An MBMS bearer has been announced to the MC application and registered within the MC MBMS user agent.

3. The MBMS bearer is de-announced by the MC service server towards the MC application.



Figure 7.2.2.2-1: MC MBMS deregistration procedure

1. The MC application deregisters the MBMS bearer with the MC MBMS user agent. No additional notification from the MC MBMS user agent is expected by the MC application related to this MBMS bearer.

2. The MC MBMS user agent interrupts all operations previously requested by the MC application for this MBMS bearer (e.g. MBMS bearer availability checking, MBMS bearer quality monitoring or consumption) and deletes any parameters and/or internal state associated to this MBMS bearer.

3. The MC MBMS user agent notifies the MC application about the deregistration result.

Post-conditions:

1. The MC application and can no more proceed any API interactions with the MC MBMS user agent related to this MBMS bearer, e.g. API interactions for media consumption.

#### 7.2.2.3 MC MBMS bearer availability

This procedure allows the MC MBMS user agent to notify the MBMS application that a registered MBMS bearer is available within the UE location.

Pre-conditions:

1. The MC application is registered towards the MC MBMS user agent.

2. An MBMS bearer has been announced to the MC application.

3. The MC application has registered this MBMS bearer within the MC MBMS user agent

4. No MBMS bearer availability notification has been sent since the MBMS bearer registration, or a MBMS bearer unavailability notification has been sent.



Figure 7.2.2.3-1: MC MBMS bearer availability procedure

1. The MC MBMS user agent continuously checks if the MBMS bearer is available at the UE location.

2. If the MBMS bearer is available, the MC MBMS user agent sends a MC MBMS bearer availability notification. If the MBMS bearer is resumed, the MC MBMS bearer availability notification contains the status of MBMS resumption.

Post-conditions:

1. The MC MBMS user agent executes the MC MBMS bearer unavailability procedure, i.e. will send a notification if the TMGI is not more present within MCCH or SC-MCCH or indicated as suspended by the MSI.

2. If MBMS bearer quality evaluation is required by the MC application, as asked during the MBMS registration procedure, the MC MBMS user agent executes the MC MBMS bearer evaluation procedure.

3. If the MBMS bearer is resumed, the MC application may send the MBMS suspension report to the MC service server indicating the MBMS resumption.

#### 7.2.2.4 MC MBMS bearer unavailability

This procedure allows the MC MBMS user agent to notify the MBMS application that a registered MBMS bearer is available within the UE location.

Pre-conditions:

1. The MC application is registered towards the MC MBMS user agent.

2. An MBMS bearer has been announced to the MC application.

3. The MC application has registered this MBMS bearer within the MC MBMS user agent

4. No MBMS bearer unavailability notification has been sent since the MBMS bearer registration, or a MBMS bearer availability notification has been sent.



Figure 7.2.2.4-1: MC MBMS bearer unavailability procedure

1. The MC MBMS user agent continuously checks if the MBMS bearer is available within the UE location, i.e. its TMGI (or one of the alternative TMGI) is listed within MCCH or SC-MCCH, or suspension indication in the MSI.

2. If the MBMS bearer is unavailable, the MC MBMS user agent sends a MC MBMS bearer unavailability notification. If the MBMS bearer is suspended, the MC MBMS bearer availability notification contains the status of MBMS suspension.

Post-conditions:

1. The MC MBMS user agent executes the MC MBMS bearer availability procedure, i.e. will send a notification if the TMGI (or one of the alternative TMGI) becomes present within MCCH or SC-MCCH.

2. The MBMS bearer quality evaluation procedure, if currently executed, is interrupted.

3. If the MBMS bearer is suspended, the MC application may send the MBMS suspension report to the MC service server indicating the MBMS suspension.

#### 7.2.2.5 MC MBMS bearer quality evaluation

This procedure allows the MC MBMS user agent to notify the MBMS application if quality of reception for a MBMS bearer is evaluated good enough for MC service media consumption.

Pre-conditions:

1. The MC application is registered towards the MC MBMS user agent.

2. An MBMS bearer has been announced to the MC application by the MC service server.

3. The MC application has registered this MBMS bearer within the MC MBMS user agent and required quality evaluation.

4. The MBMS bearer is available at the UE location.



Figure 7.2.2.5-1: MC MBMS bearer quality evaluation procedure

1. The MC MBMS user agent continuously evaluates reception for the MBMS bearer.

2. When reception quality evaluation is done and

- no notification for the reception quality evaluation has been done since the last MC MBMS bearer availability notification for this given MBMS bearer; or

- the reception quality evaluation is different to the last reception quality evaluation notification,

then, the MC MBMS user agent notifies the MC application if the reception quality evaluation is good or bad.

Note: The need for a hysteresis within the MC MBMS user agent, to avoid sending too frequent reception quality evaluation notifications is left to implementation.

Post-conditions:

1. The MC MBMS user agent keeps executing the MC MBMS bearer quality evaluation procedure.

### 7.2.3 Information flows

#### 7.2.3.1 MC MBMS bearer registration request

Table 7.2.3.1-1 describes the information flow for the MC MBMS bearer registration request.

Table 7.2.3.1-1: MC MBMS bearer registration request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| TMGI | M | TMGI information |
| List of service area identifier | M | A list of service area identifier for the applicable MBMS broadcast area |
| Frequency | O | Identification of frequency if multi carrier support is provided |
| Monitoring reception quality | O | This boolean is used to control if the MC MBMS user agent is actively monitoring the MBMS bearer quality or not |
| QCI | O | Quality class identifier (NOTE) |
| NOTE: The QCI provides an identifier about the tolerance to loss for the services to be delivered within the MBMS bearer and is used by the MC MBMS user agent to determine if the reception quality is good enough. This IE is included in the information flow if Monitoring reception quality is set to true. | | |

#### 7.2.3.2 MC MBMS bearer registration response

Table 7.2.3.2-1 describes the information flow for the MC MBMS bearer registration response.

Table 7.2.3.2-1: MC MBMS bearer registration response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| TMGI | M | TMGI information |
| Response code | M | Provides the result of the registration request: success, failure or missing parameters |
| Message | O | Additional details |

#### 7.2.3.3 MC MBMS bearer deregistration request

Table 7.2.3.3-1 describes the information flow for the MC MBMS bearer deregistration request.

Table 7.2.3.3-1: MC MBMS bearer deregistration request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| TMGI | M | TMGI information |

#### 7.2.3.4 MC MBMS bearer deregistration response

Table 7.2.3.4-1 describes the information flow for the MC MBMS bearer deregistration response.

Table 7.2.3.4-1: MC MBMS bearer deregistration response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Response code | M | Provides the result of the deregistration request: success, failure or missing parameters |
| Message | O | Additional details |

#### 7.2.3.5 MC MBMS bearer event notification

Table 7.2.3.5-1 describes the information flow for the MC MBMS bearer event notification. This information flow is used for notification about MBMS bearer availability, unavailability and quality evaluation.

Table 7.2.3.5-1: MC MBMS bearer event notification

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| TMGI | M | TMGI information |
| Event code | M | Code providing the nature of the event |
| Message | O | Additional details |
| NOTE: For notifications of MBMS bearer availability and unavailability, indication of MBMS suspension or MBMS resumption status may be included. | | |

### 7.2.4 Evaluation

This solution introduces a simple API to facilitate implementation of MBMS procedures from 3GPP TS 23.280 [3]: MBMS bearer quality detection, MBMS suspension notification, switching from MBMS bearer to unicast bearer. Complexity of the MBMS bearer quality evaluation is delegated to the MC MBMS user agent.

## 7.3 API for media reception

### 7.3.1 General

This solution covers key issues of "Announcement handling" and "ROHC and FEC handling".

This solution pre-requires that the list of announced MBMS bearers is maintained internally by the MC MBMS user agent, including MBMS SAI and frequencies information, and that a notification mechanism has been initialized during the application registration (key issue 5.5).

It introduces a set of procedureswithin the MC MBMS API to start/stop reception of a given media delivered within MBMS. This media can be: the application level control signalling, or the media stream of a group communication.

Editor's note: A procedure to notify the MC application about media reception errors is FFS.

In this solution, header decompression and FEC decoding can be performed by the MC MBMS user agent or by the MC application.

The solution proposes two methods to return the media from the MC MBMS user agent to the MC application: by providing them on a UE network interface or by returning them with a callback mechanism.

### 7.3.2 Procedures

#### 7.3.2.1 Open media

This procedure allows the consumption of a media delivered over MBMS

Pre-conditions:

1. The MC application is registered towards the MC MBMS user agent

2. A MBMS bearer has been announced to the MC application through the MBMS procedures specified in 3GPP TS 23.280 [3] and registered toward the MC MBMS user agent.

3. This MBMS bearer is available within the UE location (i.e. present within the MCCH or SC-MCCH).

4. The MC application has received a MapGroupToBearer message (see 3GPP TS 23.379 [6], table 10.10.1.1-1 for MCPTT and 3GPP TS 23.281 [4], table 7.10.1.2-1 for MCVideo), indicating a group communication delivered over this MBMS bearer, orthe announced MBMS bearer delivered the application level control signalling.



Figure 7.3.2.1-1: Open media procedure

1. The MC application asks to access to the communication with the open media request, by indicating the TMGI of the MBMS bearer and a list of media description information. If the MC application requests FEC decoding and/or header decompression, relevant FEC and ROHC information are included within the open media request; if the MC application wants the media packets to be returned by callback, the MC application provides a callback listener.

2. The MC MBMS user agent checks that the MBMS bearer is available. If the modem is not already receiving this bearer, the MC MBMS user agent asks the modem to start the reception of this given MBMS bearer.

3. When the media is available, the MC MBMS user agent sends an open media response to the MC application, including a description of the required media (after FEC decoding). If packets are returned to the MC application through a local network interface, this network interface name is included in the response.

4. The MC MBMS user agent decompresses headers of the required media if requested in step 1, performs the FEC decoding if requested in step 1, and makes the media streams available to the MC application according to the configured method for returning media with a help of a local network interface, or by returning the media packets to the callback listener provided in step 1.

Post-conditions:

1. The MC application can consume the required media, until interruption if the MBMS bearer becomes unavailable (in that case, the MC MBMS user agent is expected to send an MC MBMS bearer unavailability notification), or until it closes the media with the close media procedure.

#### 7.3.2.2 Close media

This procedure allows to stop the consumption of a media delivered over MBMS.

Pre-conditions:

1. The MC application is consuming a media, after having call the open media procedure.

2. The MC application has received a UnMapGroupToBearer message (see 3GPP TS 23.379 [6], table 10.10.1.2 -1 for MCPTT and 3GPP TS 23.281 [4], table 7.10.1.3-1 for MCVideo), indicating the end of a group communication delivered over this MBMS bearer, orthe reception quality is too bad (which can be notified by the MC MBMS user agent), and the MC application wants to switch to unicast (see procedure in clause 10.7.3.3 of 3GPP TS 23.280 [3]orsome reception/FEC decoding/header decompression errors have been notified to the MC application by the MC MBMS user agent.



Figure 7.3.2.2-1: Close media procedure

1. The MC application asks the MC MBMS user agent to stop the reception of media streams with the close media request.

2. The MC MBMS user agent stops the possible header decompression and FEC decoding.

3. The MC MBMS user agent notifies the MC application that the media has been closed.

Post-conditions:

1. The closed media streams are no more available to the MC application.

2. If no other media from this MBMS bearer are opened, the MC MBMS user agent can end the reception of this MBMS bearer.

### 7.3.3 Information flows

#### 7.3.3.1 Open media request

Table 7.3.3.1-1 describes the information flow for the open media request.

Table 7.3.3.1-1: Open media request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| TMGI | M | TMGI information of the announced MBMS bearer |
| Media description | O | Describes the multicast IP adresses and ports of the requested media. (e.g. the MBMS subchannel). |
| FEC information | O | FEC information if the MC application requests the MC MBMS user agent to perform the FEC decoding. |
| ROHC information | O | ROHC information for the requested media. Indicates if header decompression by the MC MBMS user agent is requested. |
| Callback | O | Dedicated callback listener where to return the received media packets. If this IE is not present, the media packets are returned over a local network interface. |

#### 7.3.3.2 Open media response

Table 7.3.3.2-1 describes the information flow for the open media response.

Table 7.3.3.2-1: Open media response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Media identifier | M | Identifier from the open media request |
| Response code | M | Indicates if the open media request was successful, otherwise an error code |
| Message | O | Additional details |
| Media description | O | Description of opened media, including the multicast IP address and port of the media, after FEC decoding. |
| Network interface | O | The network interface where the given media can be consumed by the MC application, if packet are forwarded to the MC application over a UE network interface according the configured method for media reception |

#### 7.3.3.3 Close media request

Table 7.3.3.3-1 describes the information flow for the close media request.

Table 7.3.3.3-1: Close media request

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Media identifier | M | Identifier from the open media request |

#### 7.3.3.4 Close media response

Table 7.3.3.4-1 describes the information flow for the close media response.

Table 7.3.3.4-1: Close media response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Media identifier | M | Identifier from the open media request |
| Response code | M | Indicates if the close media request was successful, otherwise an error code |
| Message | O | Additional details. |

### 7.3.4 Evaluation

This solution defines a simple API for media reception over MBMS. It allows FEC and ROHC to be performed either in the MC MBMS user agent or in the MC application. It allows two methods for returning media to the MC application, defined by configuration.

## 7.4 Location information API

### 7.4.1 General

This solution covers the requirements [MBMSAPI-FR-2] and [MBMSAPI-FR-3], and the key issue "5.7 location information".

### 7.4.2 Procedures

#### 7.4.2.1 Get SAI

Figure 7.4.2.1-1 illustrates the procedure of get SAI information by which the MC application queries the current SAI(s) that the MC service UE is located.

Pre-conditions:

1. The MC application has registered towards the MC MBMS user agent.



Figure 7.4.2.1-1: Get SAI procedure

1. The MC application queries the SAI information from the MC MBMS user agent.

2. The MC MBMS user agent responds with the SAI(s) of which the UE is currently located.

Post-conditions:

1. The MC application is able to report the SAI information.

#### 7.4.2.2 SAI update notification

Figure 7.4.2.2-1 illustrates the procedure of SAI update notification by which the MC application is notified about the changed SAIs upon the occurrence that the MC service UE is moving to new MBMS service area. The subscription for SAI update notification is performed with the registration of the MC application.

Pre-conditions:

1. The MC application has registered towards the MC MBMS user agent.

2. The SAI(s) where the MC service UE is located has changed when the UE is moving.



Figure 7.4.2.2-1: SAI update notification procedure

1. The MC MBMS user agent detects the SAI(s) of which the MC service UE resides on has changed.

2. The MC MBMS user agent notifies the MC application about the updated SAI information.

Post-conditions:

1. The MC application is able to report the SAI information.

#### 7.4.2.3 Get cell ID

Figure 7.4.2.3-1 illustrates how the MC application queries the MC MBMS user agent to get the current cell ID of an MC service UE.

Pre-conditions:

1. The MC application is registered at the MC MBMS user agent.



Figure 7.4.2.3-1: Get cell ID procedure

1. The MC application queries the cell ID information from the MC MBMS user agent.

2. The MC MBMS user agent responds with the cell ID where the UE is currently located.

Post-conditions:

1. The MC application may report the cell location information to the MC service server.

#### 7.4.2.4 Cell ID update notification

Figure 7.4.2.4-1 illustrates the procedure of cell ID update notification by which the MC application is notified when the MC service UE moves towards a new cell. The subscription for cell ID update notification is performed with the registration of the MC application.

Pre-conditions:

1. The MC application is registered at the MC MBMS user agent and has indicated to receive cell update notifications.

2. The cell ID where the MC service UE is located has changed when the UE is moving.



Figure 7.4.2.4-1: Cell ID update notification procedure

1. The MC MBMS user agent detects that the cell ID where the MC service UE resides has changed.

2. The MC MBMS user agent notifies the MC application about the updated cell ID information.

Post-conditions:

1. The MC application may report the cell location information to the MC service server.

### 7.4.3 Information flows

#### 7.4.3.1 Get SAI response

Table 7.4.3.1-1 describes the information flow for the get SAI response.

Table 7.4.3.1-1: Get SAI response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| SAIs | M | A list of SAIs that are available where the MC service UE is currently located |
| Result | M | Result of the API call (success or failure) |

#### 7.4.3.2 SAI update notification

Table 7.4.3.2-1 describes the information flow for the SAI update notification.

Table 7.4.3.2-1: SAI update notification

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| SAIs | M | A list of SAIs that are available where the MC service UE is currently located |

#### 7.4.3.3 Get cell ID response

Table 7.4.3.3-1 describes the information flow of the get cell ID response message.

Table 7.4.3.3-1: Get cell ID response

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Cell ID | M | Cell ID of the MC service UE |
| Result | M | Result of the API call (success or failure) |

#### 7.4.3.4 Cell ID update notification

Table 7.4.3.4-1 describes the information flow of the cell ID update notification message.

Table 7.4.3.4-1: Cell ID update notification

|  |  |  |
| --- | --- | --- |
| Information element | Status | Description |
| Cell ID | M | Cell ID of the MC service UE |

### 7.4.4 Evaluation

The solution addresses the key issue in subclause 5.7 "location information".

The solution enables the MC application to obtain the MC service UE's location information (i.e. SAI or ECGI) to be reported to the MC service server. The location information API can be used for MBSFN mode and SC-PTM mode.

## 7.5 MBMS API for MCData file distribution

### 7.5.1 General

This solution covers the key issue API for file distribution over MBMS and is related to the use case defined in subclause 6.3 Using the MBMS user services for MCData file distribution.

This solution proposes to reuse the existing MBMS API for file distribution, specified in subclause 6.2 of 3GPP TS 26.347 [9].

This solution extends the reference model defined in subclause 4.2 as follows:

- The MC MBMS user agent plays the role of the MBMS client and exposes the MBMS API for file distribution, specified in subclause 6.2 of 3GPP TS 26.347 [9] within the MC MBMS API.

- The MC application plays the role of the MBMS aware application when using the MBMS API for file distribution, specified in subclause 6.2 of 3GPP TS 26.347 [9], exposed by the MC MBMS user agent.

This solution reuses the registration and deregistration API solution from subclause 7.1.

The MC application can request the reception of a file with the startFdCapture()method and is informed when the file has been received with the fileAvailable() notification, as described in subclause 7.5.3.

### 7.5.2 Application registration and deregistration

An MBMS aware application for file distribution, needs to register to consume MBMS user services with the registerFdApp() and registerFdResponse()methods, specified in subclause 6.2.3.2 and subclause 6.2.3.3 of 3GPP TS 26.347 [9].

The registerFdApp(), registerFdResponse()and deregisterFdApp() methods provide similar procedures to the registration and deregistration API solution (in subclause 7.1) and address the same needs.

This solution proposes to not expose registerFdApp(), registerFdResponse() and deregisterFdApp() methods within the MC MBMS API, and to reuse the registration and deregistration API solutions.

An additional and optional information element can be added to the application registration request (see subclause 7.1.3.1), providing a local directory available on the device storage, which the MC application can access and where successfully collected files can be copied/moved before notifying that the file is available to the MC application.

### 7.5.3 File reception

The MC application can request the MBMS client to receive of files delivered with the MBMS download delivery method with the startFdCapture(),and fileAvailable() methods specified in subclause 6.2.3.7 and subclause 6.2.3.8 of 3GPP TS 26.347 [9].

As preconditions before calling the startFdCapture()method:

1. The MC application is registered towards the MBMS client (see subclause 7.5.2).

2. The MCData server has created a MBMS user service over xMB (step 1 of the flow in subclause 6.5.3.1).

3. The delivery of a file over a MBMS user service has been notified to the MC application (step 3 of the flow in subclause 6.5.3.1). The notification contains the MBMS user service id used for the delivery and an URI identifying the file to be received.

When calling the startFdCapture()method, the MC application provides the MBMS user service id used for the delivery and the URI identifying the file to be received. Additionally, the MC application may set the captureOnce argument to true, so that the MBMS client stops capturing new possible versions of the file once it has been received.

When the file has been received and is available to the application, the MC MBMS user agent, playing the role of the MBMS client sends a fileAvailable()notification to the MC application, indicating the file location where the MC application can find the received file.

### 7.5.4 Evaluation

This solution requires to include the MBMS API for file distribution, specified in subclause 6.2 of 3GPP TS 26.347 [9], within the MC MBMS API.

This solution requires that the MBMS user service id and the file information are provided to the MC application within new signalling messages.

This solution also requires that the MBMS user service architecture is introduced within the architecture for MCData file distribution.

The gaps in subclause A.7.1.2 and subclause A.7.1.3 need to be addressed:

1. A business agreement between the content provider and the operator allows the MCData AS to control exactly the coverage of its MBMS bearers, by using the geographical areas provided by the MCData AS to the BM-SC (see the gap in subclause A.7.1.2).

2. The QoS parameters agreed between the content provider (MCData AS) and the operator (BM-SC) matches the QoS requirements for MCData file distribution (see the gap in subclause A.7.1.3).

This solution relies fully on existing recommended capabilities.

# 8 Evaluations

## 8.1 API solution evaluation

The table 8.1-1 lists all the API solutions developed in this technical report and evaluations. The table includes the corresponding key issues and the use cases in which the API can be applied for the MBMS usage defined in 3GPP TS 23.280 [3], 3GPP TS 23.379 [6], 3GPP TS 23.281 [4] and 3GPP TS 23.282 [5].

Table 8.1-1 API solution evaluation

| API Solution | Applicable key issue(s) | Applicable use case(s) | Evaluation  (subclause reference) |
| --- | --- | --- | --- |
| 7.1 Registration and deregistration API | 5.5 Registration |  | 7.1.4 |
| 7.2 MBMS bearer registration and notification API | 5.3 Feedback on MBMS reception,  5.6 MBMS bearer quality monitoring | 6.1 MBMS bearer quality monitoring and data reception,  6.3 Switching from MBMS delivery to unicast delivery,  6.4 MBMS bearer suspension and resumption | 7.2.4 |
| 7.3 API for media reception | 5.1 Announcement handling  5.2 ROHC and FEC handling | 6.1 MBMS bearer quality monitoring and data reception,  6.3 Switching from MBMS delivery to unicast delivery | 7.3.4 |
| 7.4 Location information API | 5.7 Location information | 6.2 Location information | 7.4.4 |
| 7.5 MBMS API for MCData file distribution | 5.4 API for file distribution over MBMS | 6.5 Using the MBMS user services for MCData file distribution | 7.5.4 |

## 8.2 Review of existing MBMS API and capabilities

Gaps on applicability of xMB based MBMS user service and the MBMS API to the mission critical services are analyzed in annex A.7.

# 9 Conclusions

This technical report fulfills the objectives of the study on MBMS APIs for Mission Critical Services. The new solutions of MC MBMS API can be applied for MBMS usage on MCPTT, MCVideo and MCData services that are aligned with multicast group communications. All of the API solutions studied in this technical report will be considered for the normative work in Release 16.

The solutions in this technical report are not exhaustive and additional solutions may be considered for normative work.

Annex A:  
Analysis of capabilities provided by the MBMS user service architecture

# A.1 General

xMB (3GPP TS 29.116 [10]) and MBMS-API (3GPP TS 26.347 [9]) are interfaces allowing provisioning, ingestion and consumption of MBMS user services. The MBMS user service architecture is based on an MBMS client on the UE side and a BM-SC on the network side.



Figure A.1-1: End-to-end architecture for application service providers using eMBMS for delivery (source 3GPP TS 26.347 [9], figure 4.1-1)

MBMS user services make use of MBMS delivery methods, providing functionality such as security and key distribution, reliability control by means of forward-error-correction techniques and associated delivery procedures such as file-repair, delivery verification. Four delivery methods are defined, namely download, streaming, transparent and group communication.

The group communication delivery method is not addressed by xMB and MBMS-API, and corresponds to the transparent delivery of data transferred by a GCS-AS.

As mentioned in 3GPP TS 26.347 [9]:

*The BM-SC and UE may exchange service and content related information either over point-to-point bearers or MBMS bearers whichever is suitable. Among others, the following MBMS procedures are defined in 3GPP TS 26.346 [5]:*

*- User Service Discovery / Announcement providing service description material to be presented to the end-user as well as application parameters used in providing service content to the end-user.*

*- MBMS-based delivery of data/content from the BM-SC to the UE over IP multicast or over IP unicast.*

*- Associated Delivery functions are invoked by the UE in relation to the MBMS data transmission. The following associated delivery functions are available:*

*- File repair for download delivery method used to complement missing data.*

# A.2 Announcement / User service discovery

The list of available MBMS user services and user service bundles, along with information on the user services is announced to the UE by the BM-SC with the user service announcement. This announcement contains a list of metadata fragments providing information about, in particular: the TMGI, the MBMS service areas and frequencies where services are delivered, the delivery schedules, the forward error correction, instructions for associated unicast delivery procedures (file repair, reception report, consumption report), the media and session descriptions.

3GPP TS 26.346 [8] specified several alternative methods to deliver the announcement: in particular, over a MBMS bearer, with the MBMS download delivery method or over HTTP.

# A.3 MBMS delivery methods

## A.3.1 MBMS streaming delivery method

The purpose of the MBMS streaming delivery method is to deliver (non-mission critical) continuous multimedia data. The delivery protocol is RTP/SRTP. Streams can be protected by forward error correction with the use of Raptor scheme as defined in IETF RFC 6681 [15] for FECFRAME as defined in IETF RFC 6363 [14].

The MBMS client can be required by the announcement to perform measurements for a set of QoE metrics and report them with the reception report procedure.

Table A.3.1-1: QoE metric list (source 3GPP TS 26.347 [9], table 8.4.2)

|  |  |  |  |
| --- | --- | --- | --- |
| QoE Metric | Streaming delivery method | Download delivery method | Metric type |
| Corruption duration metric | ✓ |  | Media |
| Rebuffering duration metric | ✓ |  | Session |
| Initial buffering duration metric | ✓ |  | Session |
| Successive loss of RTP packets | ✓ |  | Media |
| Frame rate deviation | ✓ |  | Media |
| Jitter duration | ✓ |  | Media |
| Content access/Switch time | ✓ |  | Session |
| Network resource | ✓ | ✓ | Session |
| Average codec bitrate | ✓ |  | Media |
| Codec information | ✓ |  | Media |
| Loss of objects |  | ✓ | Session |
| Distribution of symbol count underrun for failed blocks |  | ✓ | Session |

## A.3.2 MBMS download delivery method

MBMS download delivery method uses the FLUTE protocol (IETF RFC 3926 [13]) when delivering content over MBMS bearers.

The purpose of this method is to deliver content in files. In the context of MBMS download, a file contains any type of MBMS data (e.g. 3GPP file (audio/video), binary data, still images, text, service announcement metadata). It can not only be used for non real time content, but also for real time content: the method can be used to delivered DASH or HLS services, by transporting their successive segments over FLUTE.

The MBMS download delivery method relies on FEC using the raptor10 scheme as defined in IETF RFC 6681 [15].

To achieve the file distribution for the small UE proportion for which the losses were too important to recover the file with FEC alone, 3GPP TS 26.346 [8] specifies the file repair procedure, allowing to repair lost or corrupted file fragments after the end of the MBMS delivery. To protect network resources, the file repair request load is to spread in time and across multiple servers.

The MBMS client can be required by the announcement to acknowledge file reception to the BM-SC, through the reception report procedure.

## A.3.3 Transparent mode (transport only)

From subclause 8B.1 specified in 3GPP TS 26.346 [8]:

*The MBMS transparent delivery method delivers application data units as part of UDP or IP flows over an MBMS bearer to the UE. This delivery method complements the download delivery method and streaming delivery method and is particularly useful for multicast and broadcast of IP-based services for which the media codecs and application protocols are defined outside of this specification.*

*The BM-SC receives Application Data Units (ADUs) from the content provider, typically provided as UDP/IP packets and forwards them to the destination multicast IP address and port number. Both IPv4 and IPv6 may be used by the transparent delivery method.*

*Transparent delivery methods may be used within MBMS User Services, where the session description is delivered as a fragment of a User Service Description, or they may be used independently, where the content provider will announce the session via external means.*

*An MBMS Transparent session may be operated in a forward-only or in a proxy mode. In the forward-only mode, the transport protocol on top of IP is opaque to the MBMS system and the session announcement is expected to be handled by the content provider itself. In the proxy mode, the UDP packet payload of the UDP streams is opaque to the MBMS session and an MBMS Client is expected to make the UDP Payloads available to an application, without further knowledge on the content.*

When the Proxy mode is used, the BM-SC shall act as the source for the multicast traffic. The SDP for the transparent delivery method shall be created by the BM-SC and provided within the announcement.

Similar to the group communication delivery method (provisioned by MB2-U, according to the GCSE architecture), the transparent delivery method (xMB-U) enables delivery of IP flows. However, additional processing may be necessary for the IP frames sent over xMB-U to create equivalence between the two methods (see subclause A.7.1).

# A.4 xMB API

The xMB API is a RESTful API that allows content providers to provision broadcast services over 3GPP networks and subsequent ingestion of service content for distribution using eMBMS.

From subclause 6.1 specified in 3GPP TS 29.116 [10]:

*The xMB-U user plane procedures cover the transmission of service data between the Content Provider to the BM-SC. Only authorized and authenticated Content Provider sources shall be able to provide user plane data over xMB-U to the BM-SC. The following data transfer modes are supported:*

*- File Push: the Content Provider uploads or transmits files to the BM-SC either as soon as they become available, or in advance.*

*- File Pull: the Content Provider makes files available prior to the session start and at least during the lifetime of a session. The BM-SC will retrieve the files when it needs to deliver them.*

*- RTP Streaming: the BM-SC establishes an RTSP session to the Content Provider and starts the streaming session to relay media streams.*

*- Transport: the BM-SC listens on one IP address and one port number to receive UDP packets.*

# A.5 MBMS API

3GPP TS 26.347 [9] defines a set of service APIs (MBMS-APIs) for MBMS services making uses of the MBMS delivery methods with the exception of the MBMS group communication delivery method. These interfaces are described with the help of IDL (Interface Definition Language). It also defines a set of interface options between the MBMS client and the application to support the transfer of user data. The programmatic library communication with the MBMS client is implementation-specific.

For each delivery method, 3GPP TS 26.347 [9] specifies a separate set of functions/notifications/internal states. The MBMS client can be used by multiple applications and must maintain multiple state machines, one per application.

The API is asynchronous, in particular for functions to open/close MBMS services.

When the RTP streams delivered by the MBMS streaming delivery method are protected with FEC, 3GPP TS 26.347 [9] recommends that:

- the MBMS client performs FEC decoding as defined in subclause 8.2.2.11 of 3GPP TS 26.346 [8]; and

- the MBMS provides to the application: a SDP that describes the decoded packet stream, the network interface from which the data stream can be received.

Received packets over MBMS for the transparent delivery method are also made available through a network interface as described for packet data interface in subclause 7.6 of 3GPP TS 26.347 [9].

# A.6 GCSE architecture

GCSE architecture and MB2 interface are specified in 3GPP TS 23.468 [7] and 3GPP TS 29.468 [11]. Their usage for MC services is specified in 3GPP TS 23.280 [3]. GCSE architecture is distinct from the MBMS user service architecture.

Over MB2-U, GCS user plane data, encapsulated in UDP and IP, are transparently transported between the GCS AS (such as an MCPTT server) and the application client at the UE. The BM-SC forwards these protocol layers transparently.

# A.7 Gap analysis

## A.7.1 xMB

### A.7.1.1 Usage of pre-established bearer

xMB does not allow the usage of pre-established bearers. When activating an MBMS bearer with a session creation request (see subclause 5.2.2.2.2 in 3GPP TS 29.116 [10]) for a RTP streaming service or a transport-only service (MBMS transparent delivery method), the content provider includes a SDP of the media streams to be ingested by the BM-SC and forwarded over MBMS.

While MB2-U is a tunnel where any type of packets can be transparently pushed, xMB requires that the list of media UDP flows is known in advanced, which is not the case for MCPTT.

### A.7.1.2 Geographical area

When activating a MBMS Bearer over xMB, the content provider includes a target geographical area within the session creation request (see subclause 5.2.2.2.2 in 3GPP TS 29.116 [10]). The geographical Area contains a list of strings. The content of each string item is left to the business agreement between the content provider and the operator. The BM-SC derives the target MBMS services area and/or cell-id from the geographical area. For mission critical services, the geographical area syntax and semantic would need to be more specified so that the GCS-AS can control exactly the coverage of its MBMS bearers.

### A.7.1.3 QoS

With MB2-C, QoS parameters (QCI, ARP) are provided by the GCS-AS. With xMB, QoS parameters are statically defined by a SLA between the operator and the content provider. Specific rules, or extensions, to xMB have to be added so that the content provider can set exactly the QoS parameters for the considered MC service.

### A.7.1.4 Header compression

xMB-U is not a tunnel and can only ingest UDP packets. Consequently, no header compressed packets can be ingested over xMB. In addition, no header compression instruction is provided over xMB-C: the BM-SC would not apply ROHC either.

### A.7.1.5 FEC

In 3GPP TS 23.280 [3], the GCS-AS can ask the BM-SC to apply AL-FEC over the ingested streams. In response, the BM-SC returns FEC information to be included within the announcement. Such mechanism does not exist in xMB: Raptor10 can be applied to RTP streams, but no FEC information is returned.

### A.7.1.6 MCData file distribution

Editor's note: MCData file distribution over MBMS is not specified yet in stage 3. Usage of download delivery method could be considered for MCData file distribution.

## A.7.2 MBMS API

### A.7.2.1 Service identity

MBMS user services are identified by their service id while MC communications over MBMS are identified by the TMGI of the MBMS bearer where they are delivered, the identifier of the media slot within the announced SDP and other information provided within MapGroupToBearer messages.

### A.7.2.2 MBMS quality detection

A set of feedbacks related to MBMS reception (suspension notification, bearer quality detection) must be reported to the MC service server. These feedbacks, in stage 3, are reported through SIP. The MBMS bearer quality reception estimation is done even if there no traffic. For MBMS user services, reception report is done over HTTP, and provides list of QoE metrics for RTP or file reception. Both the metrics and the procedure to report are not inline.

### A.7.2.3 Header decompression

There is no ROHC for MBMS user services, and consequently no header decompression capacity within the MBMS client. Moreover, the user plane interface of MBMS API is based on the usage of network interface to transfer the streams to the application. Such interface only allows the transfer of UDP flows, and header compressed packets could not be transferred to the MC service client.

### A.7.2.4 FEC

FECFRAME/Raptor10 FEC decoding is handled for the MBMS streaming delivery method by the hypothetical FEC decoder, returning a SDP of the decoded stream, and the network interface where such streams are available.

Editor's note: FEC scheme selection is currently being studied by SA4. Depending on SA4 study outputs, another FEC scheme could be specified for MC services.

### A.7.2.5 Announcement

The MBMS user service announcement provides all the necessary metadata (e.g. SDP, reception report/file repair/consumption report parameters, broadcast area and more). It is usually available over MBMS with the SACH (Service Announcement Channel). For MC services, the announcement is provided into SIP and could also be delivered within MBMS. An acknowledgement procedure is defined to inform the MC service server that the announcement has well been received.

### A.7.2.6 Location reporting

The location (MBMS SAI or ECGI) may be reported within the reception reports (there is a dedicated metric "network\_ressource", allowing to report the cell id), and with the consumption reports. It is reported only when the service is consumed. For MC service, the location is reported within a SIP message and it is decorrelated to consumption of group communications.

Annex B:  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2017-07 | - | - | - | - | - | Initial version | 0.0.0 |
| 2017-07 | SA6#18 |  |  |  |  | Implementation of the following p-CRs approved by SA6:  S6-171050, S6-171051, S6-171084, S6-171085, S6-171099, S6-171126, S6-171127, S6-171128, S6-171132. | 0.1.0 |
| 2017-10 | SA6#19 |  |  |  |  | Implementation of the following p-CRs approved by SA6: S6-171483, S6-171484, S6-171488, S6-171492 | 0.2.0 |
| 2018-01 | SA6#21 |  |  |  |  | Implementation of the following p-CRs approved by SA6:  S6-180137, S6-180139, S6-180191, S6-180192, S6-180215, S6-180216, S6-180224, S6-180225, S6-180226 | 0.3.0 |
| 2018-02 | - |  |  |  |  | Implemented editorial and formatting related corrections | 0.3.1 |
| 2018-03 | SA6#22 |  |  |  |  | Implementation of the following p-CRs approved by SA6:  S6-180397, S6-180402, S6-180463, S6-180464, S6-180465, S6-180466, S6-180468, S6-180473, S6-180483, S6-180493 | 0.4.0 |
| 2018-03 | SA#79 | SP-180160 |  |  |  | Submitted for information at SA#79 | 1.0.0 |
| 2018-04 | SA6#23 |  |  |  |  | Implementation of the following p-CRs approved by SA6:  S6-180576, S6-180635, S6-180692, S6-180709, S6-180711, S6-180731, S6-180732, S6-180738, S6-180740, S6-180741 | 1.1.0 |
| 2018-05 | SA6#24 |  |  |  |  | Implementation of the following p-CRs approved by SA6:  S6-180771, S6-180805, S6-180806, S6-180820, S6-180821, S6-180862, S6-180864, S6-180867, S6-180918, S6-180929, S6-180942, S6-180943 | 1.2.0 |
| 2018-06 | SA#80 | SP-180376 |  |  |  | Submitted for approval at SA#80 | 2.0.0 |
| 2018-06 | SA#80 | SP-180376 |  |  |  | MCC editorial update for publication after TSG SA approval (SA#80) | 16.0.0 |