ETSI TR 126 953 V16.0.0 (2020-09)

Universal Mobile Telecommunications System (UMTS);
LTE;
Interactivity support for 3GPP-based streaming and download services
(3GPP TR 26.953 version 16.0.0 Release 16)

Reference
RTR/TSGS-0426953vg00

Keywords
LTE,UMTS

ETSI

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

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

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

Intellectual Property Rights

Essential patents

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

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

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Legal Notice

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

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

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

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

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

In the 3GPP context, service interactivity refers to a class of features that enables user engagement during the consumption of a streaming or downloaded service/content on the UE, distributed over broadcast or unicast bearers. Examples of services/contents which may offer interactivity capabilities include linear/live TV services, video-on-demand programs, and pre-downloaded media content which a user can consume later on, in a time-shifted manner. For video content, for example a TV program or an advertisement, service interactivity facilitates active watching (as opposed to passive viewing) by allowing the end-user to actively interact and participate with the presented content. In the context of services delivered over PSS or MBMS, different forms of service interactivity may be possible, for example:

- Voting for a favourite performer,
- Dynamic quizzes, surveys, elections,
- Rating of a live event during a program,
- Web access to additional information related to main content,
- Online chats about actors in TV episode or movie,
- Interactive advertisements,
- eCommerce and online shopping,

and many others.

Personalized and interactive service capabilities in 3GPP streaming and download services, via unicast and/or broadcast delivery, can drive higher end-user satisfaction and loyalty to the service operator, i.e., create greater "stickiness" of the operator's service offerings such as linear TV programs, live sports events and downloadable multimedia content. It could also enable the operator to further monetize streaming services (especially when distributed over MBMS) by, for example:

- Increasing the subscriber base through premium contextual service offerings.
- Supporting on-demand information or targeted advertising via simple user interaction such as click-to-call, click-to-SMS, or click-to-Web access.

- Driving greater cellular airtime or data volume usage associated with end-user initiated traffic pertaining to interactivity, the fees for which may be borne by a 3rd-party entity such as an advertiser or content provider, as opposed to the subscriber.

Service interactivity in 3GPP streaming and download service comprises application/presentation layer functionality, pertaining to user interface and user experience afforded by the interactivity feature, as well as a transport/signaling component, responsible for the discovery, synchronization and delivery of application and media content that define the interactivity experience. While the definition of the former, application/presentation layer aspects is largely outside the scope of 3GPP specification, delivery and signaling functions for enabling interactivity are fully within 3GPP's domain. Many of the tools required to signal and deliver functional components of interactive services may already exist in 3GPP PSS and MBMS specifications, but need to be clearly understood, leveraged, and possibly extended to fully support service interactivity.

The present document examines major use cases for service interactivity in 3GPP streaming and download services, and associated requirements and potential gaps in existing PSS and MBMS service layer specifications for interactivity support. It surveys service interactivity functionality specified in broadcast TV standards, as a reference point for potential emulation by, or differentiation from, 3GPP-defined mechanisms. It concludes by identifying functional gaps in MBMS and PSS specifications to support the identified recommended requirements regarding interactivity, and summarizes the necessary capabilities to fulfill those gaps.

1 Scope

The present document covers the study of interactivity support for 3GPP-based streaming and download services, in the context of services delivery over MBMS and PSS. Topical areas addressed include the following:

- Use cases and associated assumptions, recommended requirements and gap analyses on the operation of and means to support interactivity capabilities in streaming and download services.
- End-to-end architecture and functional component models, with emphasis on unicast and broadcast DASH services.
- Differentiation between application/presentation level and transport/service layer functions in support of interactive services.
- Overview of service interactivity mechanisms in terrestrial broadcast TV services, as defined in DVB and ATSC specifications.
- Notification mechanisms to activate interactive events, either pre-scheduled or to occur dynamically and unpredictably, at precise times during consumption of a main service or program.
- Functional gaps in current MBMS and PSS application/service layer specifications to support desired interactivity functions.
- Measurement and reporting of the consumption of interactive features.

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] ETSI TS 102 796 (V1.2.1): "Hybrid Broadcast Broadband TV", November 2012.

[3] ETSI TS 102 809 (V1.2.1): "Digital Video Broadcasting (DVB); Signalling and carriage of interactive applications and services in hybrid broadcast/broadband environments", July 2013.

[4] ETSI TS 102 796 (V1.4.1): "Hybrid Broadcast Broadband TV", August 2016.

[5] DVB Blue Book A168; DVB-DASH, available at: https://www.dvb.org/standards/dvb-iptv

[6] ATSC Candidate Standard: "Application Signaling" (A/337), 20 December 2016.

[7] ATSC Candidate Standard: "ATSC 3.0 Interactive Content" (A/344), 29 December 2016.

[8] ATSC Candidate Standard, "Signaling, Delivery, Synchronization, and Error Protection", 21 September 2016.

[9] ISO/IEC 23009-1:2014: "Information technology -- Dynamic adaptive streaming over HTTP (DASH) -- Part 1: Media presentation description and segment formats".

[10] 3GPP TS 26.247: "Transparent end-to-end Packet-switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH).

[11] W3C Recommendation 28 October 2014, HTML5: "A vocabulary and associated APIs for HTML and XHTML", http://www.w3.org/TR/html5/.

[12] Piesing, Jon, "Liaison Letter on Mapping MPEG DASH Events to HTML5 Text Tracks and Cues", http://lists.w3.org/Archives/Public/public-html/2013Dec/0015.html.

[13] IETF RFC 5261, "An Extensible Markup Language (XML) Patch Operations Framework Utilizing XML Path Language (XPath) Selectors", September 2008, https://tools.ietf.org/html/rfc5261.

[14] DASH Industry Forum, "Guidelines for Implementation: DASH-IF Interoperability Points", Version 4.0, December 12, 2016, http://dashif.org/wp-content/uploads/2016/12/DASH-IF-IOP-v4.0-clean.pdf.

[15] 3GPP TR 26.848: "Multimedia Broadcast/Multicast Service (MBMS); Enhanced MBMS Operation".

[16] W3C Recommendation 27 June 2001: "XML Linking Language (XLink)" Version 1.0.

[17] ISO/IEC 23008-11: "MPEG Composition Information".

[18] 3GPP TS 26.346: "MBMS Multicast/Broadcast Service; Protocols and codecs".

[19] 3GPP TS 26.347: "MBMS URLs and APIs".

[20] 3GPP TR 26.907: "HTML5 for a new presentation layer in 3GPP services".

[21] 3GPP TS 26.142: "Dynamic and Interactive Multimedia Scenes (DIMS)".

[22] 3GPP TS 26.234: "Transparent end-to-end Packet-switched Streaming Service (PSS), Protocols and codecs".

[23] 3GPP TS 33.246: "3G Security; Security of Multimedia Broadcast/Multicast Service (MBMS)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TR 21.905 [1].

Event: Timed notification to UE software or to an application, indicating that some action is to be taken.

Hybrid application/app: A web application wrapped inside a native application. In the context of MBMS, hybrid apps comprise a category of MBMS applications ("MBMS application" is defined in TS 26.346 [18]).

Interactive service: An MBMS or PSS service characterized by the ability of users to interact with the content/program in one or both of the following ways: 1) by changing the presented content (e.g. via access to auxiliary information, change of camera angle, supplementary media content overlaid on main program, concurrent display of text with the main video, etc.); 2) by returning end-user-supplied information or -initiated action to the service provider or content provider through the unicast channel (for example to vote for a particular choice, order a product, or participate in an on-screen quiz).

Interactivity experience: The end-user experience that result from the occurrence of one or more interactivity events during the presentation of an interactive service.

Interactive media: Media content, as part of an interactive service, presented to an end-user to prompt explicit action by the user, and/or in response to user input.

Native application/app: An application developed specifically for a particular mobile device/operating system and is installed directly onto that device. Users of native apps typically download them via app stores online or the app marketplace. In the context of MBMS, native apps comprise a category of MBMS applications.

Web application/app: An Internet-enabled, client-server software application, typically written in HTML, Javascript and CSS, for which the client (or user interface) runs in the web browser. In the context of MBMS, web apps comprise a category of MBMS applications.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 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 TR 21.905 [1].

ACR	Automatic Content Recognition
AIT	Application Information Table
API	Application Programming Interface
ATSC	Advanced Television Systems Committee
AVC	Advanced Video Coding
CENC	Common Encryption
CSS	Cascading Style Sheet
DAE	Declarative Application Environment
DO	Declarative Object
DOM	Document Object Mode
DSM-CC	Digital Storage Media – Command and Control
DTV	Digital TeleVision
DTVCC	DTV Closed Caption
DVB	Digital Video Broadcasting
EBU-TT	European Broadcasting Union Timed Text
FP	FingerPrinting
HbbTV	Hybrid Broadcast Broadband TV
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
MPEG	Motion Picture Experts Group
IDTV	Integrated digital television (receiver)
ISOBMFF	International Organization for Standards, Base Media File Format
NRT	Non Real Time
OIPF	Open IPTV Forum
PVR	Persona Video Recorder
SMT	Service Map Table
STB	Set-Top Box
TDO	Triggered Declarative Object
TPT	TDO Parameters Table
UDO	Unbound Declarative Object
VoD	Video on Demand
WM	WaterMark or WaterMarking
XHTML	Extensible HyperText Markup Language
XML	EXtensible Markup Language

4 Interactivity Support for 3GPP-Based Streaming and Download Services (IS3)

4.1 Introduction

The sub-clauses in this section describe a set of service interactivity specific use cases and related analysis for several of them with regards to scene updates, working assumptions, recommended requirements, and gap analysis pertaining to those recommended requirements.

4.2 Use Cases, Working Assumptions, Recommended Requirements and Gap Analysis

4.2.1 Use Cases

4.2.1.1 Use Case #1: Mobile TV with Auxiliary Data and User Interactivity

Frank is watching a TV talent show program "America's Top Singers" on his UE. Delivered along with the broadcast content is auxiliary data content including web links to access additional information on the performer's background and competition status, on-screen display for real-time user feedback opportunities and results. The auxiliary contents are synchronized with the A/V stream and rendered as a side-bar along with the main program by the client application on the UE. The auxiliary data is updated at different times during the main program for user engagement, such as display of buttons and links that can be selected by the user to obtain additional multimedia information on the chosen singer. After the performances for the program have completed, audience participation via "vote buttons" alongside the performer's names are displayed to enable viewer selection of their favourite performer among the competitors. The user's choice is sent to the program's vote compilation server, and during the voting period, tallied results are displayed in real-time. After the allowed time period for user interaction has elapsed, and upon the show host announcement of the evening's results, the final vote results are displayed to indicate the 1st, 2nd and 3rd place performers of the evening. The winner, Beyondme, walks up to the podium in tears of joy as the audience wildly applauses, and the host makes the obligatory congratulatory remarks and reiterates that "America's Top Singers" is the top-rated TV talent show in the country.

4.2.1.2 Use Case #2: Click for Info

The viewer of the "My European Vacation with Tom" video service is able to interact with the content by clicking on a combination of pop-up buttons and web links. Some of these buttons and links are statically displayed alongside the main display throughput the program, while others appear and later disappear dynamically, at specific times during the travelogue program to obtain more information related to the specific cities and tourist sites featured during the program segment. The returned information may contain advertisements on cruises and vacation packages.

4.2.1.3 Use Case #3: Dynamic Interactive Ads During Live Sports Event

Live streaming of the 2020 Superior Bowl football game is offered by mobile operator Horizon. The game features the Patriots with 42 year-old quarterback T. Bradley and the Broncos with 43-year old quarterback P. Manny. Early in the 1st quarter, a vicious sack of Manny leaves him unconscious on the field. As a time-out is called, an interactive wine commercial is displayed to viewers with an on-screen link that enables user access to more information on the wines produced by the sponsor, along with a chance to enter a drawing to win a winery tour. Later in the game, with 20 sec remaining in the 4th quarter, with the score tied at 20-20 tie, the Patriots have the ball and Bradley goes for an unexpected quarterback sneak and scores the winning touchdown. The fans go crazy, and as the field is swamped with players and fans, the service provider decides to interrupt the broadcast with another interactive ad.

4.2.1.4 Use Case #4: Dynamic and Personalized Interactive Ads During Live Sports Event

Jack and Jill are watching the same football game as described in the previous use case. During the aforementioned injury timeout in the 1st quarter, a personalized interactive ad is displayed on Jack's screen inviting him to view a sport car commercial at the end of which he is asked to answer three questions and is notified that he will be entered in a drawing to win that car. At the same time, a different customized ad is presented on Jill's screen on women's couture, for which she is invited to pick her favourite dresses among those displayed and submit her vote online. During the second aforementioned game interruption, another set of personalized and interactive ads are presented on Jack's and Jill's UEs.

4.2.1.5 Use Case #5: Measurement and Reporting of Interactivity Usage

The MBMS operator "Colossus Wireless" offers a number of MBMS User Services which are associated with service interactivity, such as the display of targeted ads for user engagement during timeouts or other dynamic and unscheduled times of live sports events, interactive voting of a favourite performer during talent show programs, opportunities for interactive purchasing of merchandise during the main program, display of links to external content related to the

current portion of the main program, etc. *Colossus Wireless* itself, and/or on behalf of a 3rd-party entity, wishes to obtain information, in a secure manner, regarding end-user usage of and engagement with the interactivity-related display content, or other forms of user engagement during the interactivity event, subject to meeting service subscription related terms and/or regulatory requirements pertaining to user privacy. *Colossus Wireless* would like to implement an interactivity usage measurement and reporting solution that will enable simple device logging functionality, i.e. the chosen UEs for reporting will simply log sequences of raw event data, while a network server can perform offline processing of reported data to correlate these events and extract usage statistics, thereby minimizing the device complexity. In addition, the MBMS operator intends to specify the time occurrences of interactivity usage reporting to coincide with off-peak network times, to minimize the impact of unicast network load from such reporting. *Colossus Wireless* will define the specific interactivity-related usage metrics to be logged, and control the user device population which will generate the reports. For the latter purpose, it intends for the reporting to involve either random selection of user devices, or it will define a specific group of devices from which the reports will be sent.

4.2.2 Working Assumptions

NOTE: The tentatively agreed working assumptions as shown below are for further study, with the intent to represent the operational environment of service interactivity as described by the use cases in clause 2, as opposed to solution framework. Additional working assumptions are expected to be added to this clause.

The following working assumptions are applicable to the use cases in this clause:

- Auxiliary data components associated with the interactive main service/program are carried over one or more delivery sessions (e.g. MBMS download sessions, PSS sessions).
- Information about the user, such as a profile, can be used to enable a personalized interactivity experience, for example personalized offers or ads displayed during the main program.
- Interactivity usage information is expected to be useful to, and/or required by the 3GPP service provider or a 3rd-party entity, for the following purposes (non-exhaustive list):
  - deriving statistics on the amount of viewing of interactive advertisements;
  - determining time durations users spent/engaged with interactive content;
  - measuring the number of click-throughs of embedded links in the displayed interactivity content;
  - counting the number of purchases and possible deriving the monetary value of these purchases, associated with the interactivity event;
  - associating user demographic information with the interactivity usage, in accordance to user-privacy related subscription terms or regulatory requirements.
- Interactivity usage information may affect advertisement revenues for the 3GPP service operator or 3rd party content provider. It could also assist determination of the popularity of an interactive advertisement or other interactivity display content associated with the main program or MBMS service, as well as provide an indication of the effectiveness of an interactive application.
- The 3GPP service operator has specific objectives for the interactivity usage measurement and reporting solution regarding simplification of device logging functionality, occurrence time and duration of interactivity usage reporting sessions, and the device population from which the reports are to be collected.
- UE collection and reporting of interactivity usage data, and the use of that data by the 3GPP service provider or 3rd-party entity, are restricted to the MBMS User Service to which the interactivity events pertain, and assumes that the user has explicitly opted in to such interactivity usage collection and reporting.
- The 3GPP operator, and on behalf of the end user, wishes to ensure the secure storage of interactivity usage information on the user device, and secure transmission of that information to the network.

4.2.3 Use Case Analysis

4.2.3.1 Scene Update Processing

In the following discussion on the use of scene updates to support interactivity use cases, MPEG Composition Information (CI) [17] is cited as an example format for describing scene updates. The MPEG CI document will require processing by a Javascript or a native CI engine. In the latter case, the CI engine is then a separate processor from the web runtime engine. In the former case, the Javascript for processing the CI documents is delivered as part of the presentation (just as is the case for the DASH MPD processor). The use of other formats for describing scene updates, including proprietary formats such as a Javascript Framework, may also be used for this purpose.

4.2.3.2 Use Case #1: Mobile TV with Auxiliary Data and User Interactivity

In this use case, a composite scene with a main video and side content is used. The side content is updated throughout the lifetime of the program. Interactivity in form of voting is offered at specific points of the presentation.

The following HTML5 document provides an example of such presentation when authored in HTML5.

```
<!DOCTYPE html>

<head>
</head>
<body>
<div id="mainContent">
  <video id="mainVideo"></video>
</div>
<div id="sideContent">
<!-- side content -->
</div>
</body>
</html>
```

Interactivity with the server is performed using HTML input elements and events and is transmitted to through HTTP using XmlHttpRequest API.

Scene updates are delivered separately. The format of scene update information/document may either be proprietary or an existing standard such as MPEG CI [17] may be used for the purpose.

If MPEG CI is used as scene update format, the scene update to address the current use case might look as follows:

```
<CI>
   <view id="sideContent">
      <area id="side" style="visibility:visible;" begin="2015-08-26T14:00:00" refDiv="sideContent"/>
      <area id="side" style="visibility:hidden;" begin="2015-08-26T14:30:00" refDiv="sideContent"/>
   </view>
</CI>
```

The example shows how the side content is first shown and then hidden after 30 minutes. That side content element may contain all the interactivity information, e.g. to perform voting and to display side information about the main show.

4.2.3.3 Use Case #2: Click for Info

Similar to the above example, this use cases can be addressed through timing the appearance and hiding of content in " "div" elements of the HTML5. The scene update information will provide CSS attribute modifications to be applied to the referenced "div"

4.2.3.4 Use Case #3: Dynamic Interactive Ads During Live Sports Event

This use cases is about unpredicted events that initiate customized ad insertion. This use case can be addressed by issuing a scene update and delivering the scene update document to the receivers over MBMS, with clear identification to accelerate retrieval and processing at the UE.

Upon reception of the scene update, the document will contain information to change the video source to an ad by pointing to the MPD of the ad. The resolution of the MPD URL may be used to serve custom ads.

If MPEG CI is used as a scene update format, the scene update document might look as follows:

```
<CI>
   <view id="sideContent">
      <mediaSync begin="2015-08-26T14:50:10" refId="mainContent">
          <sourceList>
              <mediaSrc mimeType="application/mpd+xml">http://www.example.com/ad.mpd</mediaSrc>
          </sourceList>
      <mediaSync/>
   </view>
</CI>
```

4.2.3.5 Use Case #4: Dynamic and Personalized Interactive Ads During Live Sports Event

This use case is about customized ads based on user profiles. HTML5 can build this logic into scripting or address resolution.

4.2.4 Recommended Requirements

The following recommended requirements from the service and transport layers perspective are derived from the interactivity use cases as described in the present document and summarized above in clause 2.1.

4.2.4.1 General

The following recommended requirements are not specific to any use case, but are applicable to all of them in order to support service interactivity:

- An interactive service experience will be available to the user while consuming either a live streaming, on-demand streaming, or non-real-time (NRT) service or program delivered via unicast or broadcast transport.
- It ought to be possible for an interactivity application, whose logic is executed by the interactivity agent, to be implemented as any of the following types:
  - a *native application*, written for a certain mobile device or platform,
  - a *web application*, written in HTML/Javascript/CSS, downloaded from a web site, and runs in the device's web browser, or
  - a *hybrid application* in the form of a web app wrapped inside a native container which provides access to native platform features.
- It is possible for the downloaded interactivity content to be cached in the UE, to be activated or displayed later on, during the presentation of the service or program for which interactivity is enabled.
- There is a means to uniquely identify an interactivity application, as one of the content components of an interactivity-enabled User Service associated with a specific interactivity use case, from other content files associated with interactivity support.
- Similar to the interactivity application, it is possible to uniquely identify among interactivity content other interactivity related files such as application data and interactivity media such as video clips, images or text files, to be played out during the interactivity event.

- It is expected that a signaling mechanism will be available to launch the execution or display of interactivity related content, in synchrony with interactivity events in the main program whose occurrence(s) may be scheduled or unscheduled.
- It ought to be possible for the 3GPP service provider to obtain reports of interactivity event-related usage information from user devices, during interactivity usage reporting sessions as defined by the 3GPP service provider.
- It ought to be possible for the 3GPP service provider to specify the parameters to be contained in the interactivity usage reports sent by user devices.
- It ought to be possible for the 3GPP service provider to control the occurrence times of interactivity usage reporting by user devices.
- It ought to be possible for user-privacy requirements, including explicit opt-in by the user and anonymity of user identity, to be fulfilled in the collection and reporting of service interactivity related usage.
- It ought to be possible for interactivity usage information to be securely stored on the user device, and to be securely transmitted from the user device to the network.

4.2.4.2 Use Case Specific Recommended Requirements

The following recommended requirements pertaining to interactivity support are derived from the existing use cases.

- From "Mobile TV with Auxiliary Data and User Interactivity":
  - Content intended for real-time and program-synchronized interactive display is assumed to be available to be downloaded and cached in the MBMS receiver for rendering later on during the main program at the appropriate time instances.
- From "Click for Info":
  - It is possible for service interactivity to be supported for NRT broadcast services.
  - Service interactivity experiences associated with NRT service/content presentation is capable of being personalized for different users.
- From "Interactive Ads in Live Events", both with and without personalization:
  - It is possible for service interactivity to occur in advertisements, as a form of content, to be presented during rendering of live services/programs, and whose time of incidence can be precisely synchronized with dynamic and unscheduled occurrences during the live service/program, for example, an injury time-out during a football game.
  - It is expected that personalization of the interactivity experience associated with an interactive ad can be provided to the user.
- From "Measurement and Reporting of Interactivity Usage":
  - It ought to be possible for the 3GPP service provider to define the following parameters contained in the interactivity usage reports: amount of viewing of interactive advertisements, time duration of user engagement with the interactivity content, number of click-throughs by the user of embedded links in the displayed interactivity content, number of purchases (and possibly associated monetary value of those purchases) associated with interactivity events, user demographic information associated with interactivity usage reports (subject to fulfilling user opt-in requirements), etc.
  - It ought to be possible for the 3GPP service provider, in the interactivity usage reporting sessions it defines, to specify a) the reporting time window(s) during which user devices are expected to upload interactivity usage information, b) the specific interactivity events to be reported, and c) selection criteria for reporting devices.
  - It ought to be possible for the 3GPP service provider to specify two types of interactivity usage reporting sessions:

a) *Randomly-sampled session*. In this type of reporting session, the device decides via a random sampling method whether or not it ought to participate in the reporting of interactivity usage. For example, the 3GPP service

provider may include a "sample percentage" value in service announcement signaling, similar to such parameter in the Associated Delivery Procedure Description (ADPD) fragment in MBMS, to effectively specify the target percentage of UEs to perform interactivity usage reporting.

b) *Targeted group session*. In this type of reporting session, it is assumed that demographic information of each end-user of the device is known to the service operator. The method to collect such user data is not expected to rely on the user providing this information outside of the application or service that intends to make use of the data. For example, each user may be associated with a certain identifier, such as a Group ID which may pertain to age, race, sex, education, income, residential community type, etc. (subject to meeting service subscription related terms and/or regulatory requirements on user privacy). Indication of the Group ID affiliated with a device may be provided inside the application or the service, or via a notification mechanism (e.g. SMS), or through unicast interaction between the UE and a network server. Only those devices whose local Group ID matches the Group ID value contained in service discovery/announcement information, for example the MBMS USD, will be required to report interactivity-related usage.

4.2.5 Gap Analysis and Evaluation

The text below in clause 2.1 identifies and evaluates potential deficiencies in TS 26.247 and TS 26.346 for supporting the recommended requirements listed in clause 4.2.4 of TS 26.953.

4.2.5.1 Gap Analysis of Interactivity Use Cases and Derived Requirements

4.2.5.1.1 Notification of Interactivity Incidences

The use cases in clause 4.2 describe the appearance of an overlaid or adjacent display, relative to the main content, which contains interactive user interface (UI) elements, such as buttons, links, icons or forms. Such rendering of auxiliary media (e.g., a banner ad, image or video clip) with embedded UI elements are intended to occur at precise times during the presentation of the main content. For example, for a pre-recorded content item such as a TV episode, the interactivity related display is expected to appear at designated time slots during the program, such as at 10 or 15-minute intervals often associated with an ad break, or at other pre-designated times during the main content. For a live event such as a football game, car race, or talent competition show, interactive display/UI elements are expected to occur dynamically and at unpredicted times during the main program, for example during the incidence of a player injury, auto crash, or live voting for a favourite performer. Tight synchronization between the main content and the auxiliary, interactivity content will be possible. In TS 26.346 [18], there is no definition of a notification mechanism that could dynamically cause the interactivity-enabled MBMS application to perform application-specific interactivity tasks at those specific times. For DASH-over-MBMS services, it may be possible to use DASH Events defined in TS 26.247 [10] as the interactivity notification mechanism. However, at this time, there is no defined API exposed by the DASH client, to enable an interactivity-enabled application or its user agent to register for callbacks, in order to obtain scheme-specific Event streams pertaining to interactivity notification messages.

4.2.5.1.2 Personalization of Interactive User Experience

The use case "Dynamic and Personalized Interactive Ads during Live Sports Event", as specified in clause 4.2.1.4 of the present document, describes the presentation of interactive advertisements at arbitrary, non-scheduled times during a live sports event. The derived requirement from this use case is that it's possible not only for the ad itself, but also the interactivity UI and experience associated with the ad, to be personalizable for a specific user or user class. The working assumption in clause 4.2.2, applicable to this use case, is that information about the user, such as a profile, can be used to enable a personalized interactivity experience. TS 26.346 [18] specifies certain capabilities for targeted content reception, such as by the user's location or group affiliation. However, it's unclear whether and how such general-purpose targeting or personalization of broadcast content reception capability can lead to or enable the occurrence of a personalized interactivity experience. It ought to be further studied whether MBMS or PSS service layer mechanisms might be defined to enable personalized interactivity in the strictly one-way, broadcast service delivery context. For example, in the case of broadcast DASH, whether the interactivity notification mechanism conveyed by DASH Event messages can be further leveraged to support personalization of the interactivity occurrences.

4.2.5.1.3 Differentiating Contents in Bundled Delivery of Application Content

As indicated in the general recommended requirements in clause 4.2.4.1, the interactivity application itself, for example a Javascript document for Web app-based interactivity application, may be bundled for delivery along with other contents associated with the interactivity app, such as media files to be rendered during the interactivity event. There

ought to be a means to uniquely identify the interactivity app from other interactivity content items, so that it can be launched in the UE upon reception, to in turn execute the interactivity tasks for which it was designed, such as acquiring and rendering interactivity media asset at specific times. For file delivery, although multipart MIME is defined as the method for bundling related files, there is no explicit mechanism defined on identifying the interactivity application among the bundled content items. A means ought to be defined on explicit signaling or implicit identification of the interactivity application in the delivery package comprising multiple interactivity-related content items.

4.2.5.1.4 Measurement and Reporting of Interactivity Usage

There is no specification of interactivity-related usage measurement and reporting functionality in the existing MBMS and PSS service layer specifications. The closest functionality to such interactivity usage measurement and reporting are the Reception Reporting and Consumption Reporting mechanisms defined in the MBMS USD, in TS 26.346 [18], in conjunction with the related signaling information contained in the ADPD. However, those measurement and reporting procedures pertain to the MBMS User Service itself, and not to auxiliary, service interactivity-related content associated with the main service or program. Signaling will need to be provided to the user device to support interactivity usage measurement and reporting. Such signaling is expected to specify the parameters of interactivity events and their usage to be collected by user devices, syntax and semantics of interactivity usage reports to be sent to the network, and metadata to control the reporting by the entirety or a subset of user devices. Signaling of those devices to perform reporting might indicate random sampling, or explicit designation, and will define the occurrence time and duration of the reporting sessions.

Towards meeting service subscription terms and/or regulatory requirements regarding user privacy, there ought be a means for ensuring that interactivity usage data can be securely stored on the user device.

4.2.5.1.5 Gap Analysis Summary

In summary, the following potential gaps in the MBMS and PSS service layer specifications are identified:

- There is need for a notification mechanism to signal the impending occurrence of either a scheduled or unscheduled event upon which time an interactivity experience is expected to be provided to the user.
- In relation to the notification mechanism, a means ought to be devised for an interactivity-aware application to be informed about the impending occurrence of, and obtain relevant information for, a service interactivity event, in order to provide the appropriate interactivity experience to the user during the interactivity event.
- A signaling mechanism ought to be provided to enable customized/personalized interactivity experiences to be delivered to different users, in a pure broadcast service/content delivery context, i.e., without requiring unicast transactions between the interactivity application and a network server.
- A method should be defined, in the case of bundled delivery of interactivity-related content items, to identify the interactivity application from interactivity assets contained in the bundle.
- A method ought to be defined in the PSS and MBMS user layer specifications, for example TS 26.234 [22] and TS 26.346 [18], to support service interactivity usage reporting.
- The protocol and message format for exchanging interactivity usage reports and their acknowledgments ought to be defined.
- The information to be provided to UEs on such reporting is expected to include (non-exhaustive list):
  - Parameters associated with interactivity events and their usage to be collected by UEs;
  - Syntax and semantics of the interactivity usage reports to be sent by the UE to the network;
  - Metadata to control actual interactivity usage reporting, by target percentage of devices to perform reporting;
  - Parameters on random sampling or explicit designation of devices for sending reports during each reporting session;
  - Indication of occurrence time and duration of interactivity usage reporting sessions.
- A means ought be provided to inform the user of the service provider's intention to collect and report his/her interactive engagement with the user service, and to enable explicit user opt-in to such data collection and reporting.

- A means ought be defined to enable secure storage of interactivity usage data on the user device.

5 Architecture Models for 3GPP Service Interactivity

5.1 DASH Service Delivery

5.1.1 General

The reference architectures for service interactivity are specific to unicast and broadcast delivery of DASH-formatted streaming services. Emphasis of these architecture is on the transport and signaling functionality at the service layer in support of interactivity. In particular, DASH Events, as defined in the MPEG DASH standard, ISO/ISC 23009-1 [9] could serve as an appropriate form of dynamic notification mechanism to initiate the execution of service application logic pertaining to service interactivity, and is assumed as the interactivity event notification mechanism in the architecture and interaction diagrams in clauses 5.1.2 and 5.1.3. Examples of service interactivity are captured in the use case descriptions in clause 4.2.

The architecture models below are exemplary in depicting interactivity in the context of DASH service delivery and assumes the use of DASH Events as the interactivity notification. Other reference architecture models and event notification mechanisms are not precluded.

5.1.2 Unicast DASH

Figure 5.1 depicts the proposed system architecture and high level sequence flow for unicast/HTTP delivery of DASH streaming services with interactivity support. Optional message steps are shown by dashed lines, and optional functionality in support of interactivity are shown inside dashed boxes. The Multimedia Framework is a software framework in the UE that handles media delivered through a network. It may provide built-in software-based codecs for popular media formats, and may also support integration with hardware codecs. The Multimedia Framework may support session management, time-synchronized rendering, transport control, and DRM. An example is the multimedia framework provided by the Android operating system. The Interactivity System collectively represents the network-side functionality that enables the desired interactive service experience by communicating with, and delivering interactivity application data and interactivity media to, the UE.

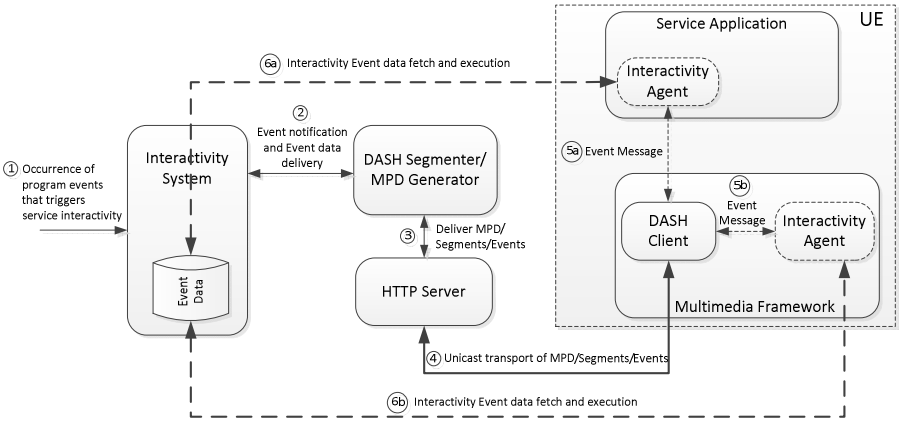

Figure 5.1: Service Interactivity Architecture for Unicast DASH

A high-level message sequence for interactivity event occurrence during a unicast streaming program, assuming the use of DASH Events as the interactivity event notification mechanism, is as follows:

1) Detection of an occurrence of a program-specific event, such as an injury time-out during a live football game, provides indication to the Interactivity System in the network that an interactivity notification and related event data are to be sent to the interactivity agent function of the service application in the UE.
2) The Interactivity System will produce related interactivity event data and pass that information to the DASH Encoder/Segmenter.

3) and 4) The DASH Segmenter/Encoder will create one or more DASH Event messages as defined in ISO/IEC 23009-1 [9], and send those either as MPD Events, or inband event messages together with the Segments (in the Event Message box 'emsg') to the DASH client, via the HTTP Server.

5a) or 5b) The DASH client delivers the Event message to the Interactivity Agent function residing either in the Multimedia Framework or in the Service Application.

6a) or 6b) (Optional) The Interactivity Agent (in the Multimedia Framework or in the Service Application) may fetch additional interactivity event data from the Interactivity System to execute the interactivity application logic, in turn creating the interactivity experience provided to the end user.

The required synchronization for the display of interactivity-related media information to the user, relative to the main program, is handled by the Interactivity Agent in the device in conjunction with the Interactivity System. The start of the interactivity event and the sequence of scenes displayed in the interactivity experience are supported by the dynamic, real-time delivery of the associated DASH Event messages to the Interactivity Agent. The DASH client is not involved in the processing of the Event messages pertaining to interactivity, and merely transfers that data as an opaque object to the Agent. The timing information and message data carried in DASH Event messages enable the Interactivity Agent to execute the interactivity logic and to display interactivity media at precisely the right times.

5.1.3 Broadcast DASH

Figure 5.2 depicts the proposed system architecture and high level sequence flow for broadcast delivery of DASH streaming services with interactivity support. Optional message steps are shown by dashed lines, and optional functionality in support of interactivity are shown inside dashed boxes.

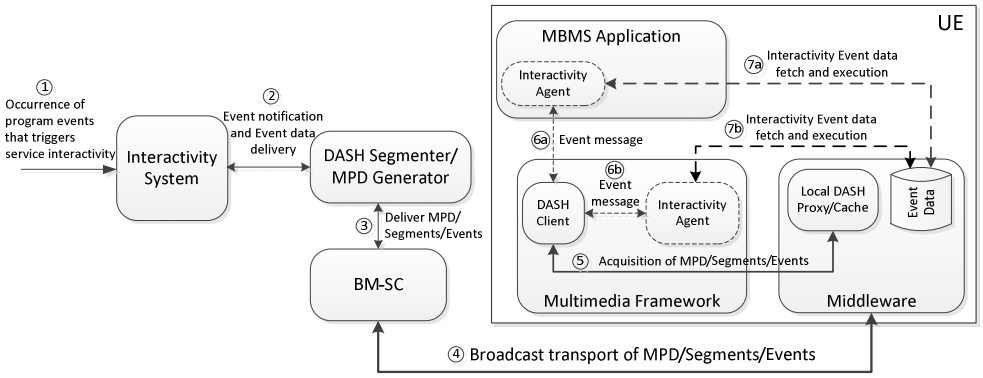

Figure 5.2: Service Interactivity Architecture for Broadcast DASH

A high level message sequence for the interactivity event occurrence during a broadcast streaming program, assuming the use of DASH Events as the interactivity event notification mechanism, is as follows:

1) and 2) Same as steps 1 and 2 in previous call flow.

3) and 4) The DASH Segmenter/Encoder will create one or more DASH Event messages as defined in ISO/IEC 23009-1 [9], and send those either as MPD Events, or inband event messages together with the Segments (in the Event Message box 'emsg') to the DASH client, via the BM-SC.

5) The DASH client retrieves MPD and Segments from the local HTTP proxy/cache in the UE.

6a) or 6b) The DASH client delivers the Event message to the Interactivity Agent function residing in either the MBMS Application or in the Multimedia Framework, whereupon the interactivity application logic may be executed by the Interactivity Agent in producing the interactivity experience for the end user.

7a) or 7b) (Optional) The Interactivity Agent in the MBMS Application or Multimedia Framework may fetch additional interactivity event data from the Middleware to execute the interactivity application logic, in turn producing the interactivity experience for the end user.

6 Interactivity Mechanisms in Broadcast and Broadband TV

6.0 General

This clause contains descriptions of interactive service framework and mechanisms as defined in DVB/HbbTV and ATSC specifications

6.1 The HbbTV / DVB Interactive Environment

6.1.1 Introduction

The HbbTV® specification, ETSI TS 102 796 [4] provides a platform for signalling, transport, and presentation of enhanced and interactive applications designed to run on hybrid terminals that include both a DVB compliant broadcast connection and a broadband connection. The HbbTV platform is open and is not based on a single controlling authority or aggregator; so services and content from many different and independent providers are accessible by the same terminal. Figure 1 in TS 102 796 [4] provides the system overview of HbbTV.

Standard functions of the terminal are available to all applications; sensitive functions of the terminal are available only to trusted applications. HbbTV is applicable to various types of terminals, including IDTVs, STBs and PVRs. Services and content may be protected. Both broadcast-related and broadcast-independent applications are supported. Broadcast applications can be presented on terminals which are not connected to broadband.

In the context of HbbTV, the main uses of the broadcast connection are the following:

- Transmission of broadcast TV, radio and data services;
- Signalling of broadcast-related applications;
- Transport of broadcast-related applications and associated data;
- Synchronization of applications and TV/radio/data services.

The main uses of the broadband connection are the following:

- Carriage of both on-demand and live content;
- Transport of broadcast-related and broadcast-independent applications and associated data;
- Exchange of information between applications and application servers.

6.1.2 DVB Signalling and Carriage of Interactive Applications

The DVB specification for application signaling and carriage, ETSI TS 102 809 [3] provides a framework for the signalling and carriage of interactive applications or services in both broadcast and broadband networks, covering the following aspects:

- Signalling of interactive applications or services:
    - This includes how the receiver identifies the applications associated with a service and finds the locations from which to retrieve them. Signalling is included that enables the broadcast service provider to manage the lifecycles of applications, and that enables the receiver to identify the sources of broadcast data required by the applications of a service. All application signalling is carried in the Application Information Table (AIT), which is carried in the PMT of the broadcast stream, in an elementary stream of private sections. All application signalling is carried in the Application Information Table (AIT), which is carried in the PMT of the broadcast stream, in an elementary stream of private sections. The XML form of the AIT is used for application signalling on broadband networks.
- Distributing the file resources of interactive applications or services:

  - Carriage of file resources is specified for two cases: MPEG-2 DSM-CC Object Carousel for broadcast carriage, and HTTP 1.1 for carriage on broadband networks.

- Synchronizing interactive applications or services to video or audio content

  Synchronization is carried out by the use of DSMCC stream events, which can comprise either "do-it-now" events for immediate activation, or stream events according to DVB timeline, for better timing accuracy. An XML equivalent of DSMCC stream event is defined, for usage on broadband networks.

- Referencing video, audio or subtitle content from interactive applications or services

  The URL form "dvb:" has been defined for referencing DVB services. The DVB application signalling specification is independent of any particular technology for interactive applications or services. It enables a wide range of different application models depending on which of the optional features are selected for the respective application environment.

6.1.3 HbbTV Platform Characteristics

Figure 2 of TS 102 796 [4] provides an overview of the functional components of the HbbTV terminal. HbbTV applications are presented by an HTML/JavaScript browser. The terminal browser environment is based on:

- OIPF Release 2:
  - Volume 5 - Declarative Application Environment
  - Volume 7 – Authentication, service protection and content protection
- TV functionality JavaScript API (OIPF Volume 5)
- CE-HTML (CEA-2014) (via OIPF Volume 5)
- W3C DOM2, CSS2, XHTML (via OIPF Volume 5 and CEA-2014)

The supported media formats are summarized by:

- OIPF Release 2, Volume 2 – Media formats, which mandates support for H.264/AVC video and HE-AAC audio, with many more optional formats defined;
- MPEG-DASH and –CENC;
- ISOBMFF live profile (applied to both on-demand and live content).

6.1.4 HbbTV Specification Evolution

The first version (V1.1.1) of the HbbTV specification was published in June 2010. The specification was revised in November 2012 and published by ETSI as TS 102 796 [2] V1.2.1. This version is also commonly referred to as "HbbTV 1.5", and it is the basis for all current HbbTV deployments. A further revision was published recently by HbbTV in February 2015, and is commonly known as "HbbTV 2.0", with the latest version represented by TS 102 796 V1.4.1 [4]. The corresponding ETSI specification revision is under way.

The major new features of HbbTV V2.0 are:

- Support for companion screens (tablets or phones) and their synchronization to broadcast delivered content;
- Privacy, based on W3C "do not track";
- Subtitles for broadband delivered content, based on EBU-TT-D;
- Interoperation with CI Plus V1.4;
- Push VoD;
- Technology updates, in particular:
  - HTML5;

- Addition of HEVC video content;
- MPEG-DASH delivered content according to DVB-DASH [5] (ETSI equivalent expected to be published soon).

HbbTV 2.0 compliant receivers are expected to start appearing in the market in 2016.

6.2 ATSC Service Interactivity

6.2.1 Introduction to ATSC

ATSC, or Advanced Television Systems Committee, Inc., is an international, non-profit organization which develops standards for digital television transmission over terrestrial, cable, and satellite networks. It was formed in 1982 by the member organizations of the Joint Committee on InterSociety Coordination (JCIC): the Electronic Industries Association (EIA), the Institute of Electrical and Electronic Engineers (IEEE), the National Association of Broadcasters (NAB), the National Cable Telecommunications Association (NCTA), and the Society of Motion Picture and Television Engineers (SMPTE). Its current member organizations represent the broadcast, broadcast equipment, motion picture, consumer electronics, computer, cable, satellite, and semiconductor industries.

ATSC is developing ATSC 3.0 as the next generation ATSC Broadcast TV transmission standard for use over terrestrial broadcast, cable and satellite networks. At the time of completion of this Technical report, it is expected that ATSC 3.0 systems will be deployed in South Korea in 2017, within several years in the USA after the FCC incentive spectrum auction and subsequent channel re-packing, and further on in the future in other major North American countries such as Canada and Mexico. ATSC 3.0 comprises a family of 19 individual specifications which include physical layer, IP-based transport protocols, service and application level signaling, content formats for carriage of streaming media services, security and content protection, and a W3C-compliant and Web application based runtime environment.

6.2.2 ATSC Service Interactivity Enabling Functionality

6.2.2.0 General

Service interactivity related functionality, or more precisely, service interactivity *enablers*, are defined in two separate ATSC 3.0 standards: A/337 "Application Signaling" [6] and A/344 "ATSC 3.0 Interactive Content" [7].

A/337 [6] defines signaling of the delivery method and properties of broadcaster applications bound to linear services (e.g., linear TV), and synchronization of application-initiated actions to the underlying audio/video content. Broadcaster applications are Web/HTML5 applications provided by TV broadcasters and associated with TV programs transmitted by those broadcasters for providing interactive features to the end-user – for example, synchronized interactive displays and targeted interactive advertising. Application-initiated actions refer to time-specific functionality defined by the broadcaster application logic – for example, the display of certain media files or advertisements, the incidents of which may be pre-scheduled or occur dynamically and unexpectedly in time. The occurrences of these actions are triggered by notifications referred to as "Events".

A/344 [7] defines the details of a W3C-compliant User Agent (i.e., web browser) based execution environment that enables a broadcaster applications to run. Such application may employ graphical capabilities of the ATSC 3.0 receiver to render the user interface or access certain resources or information related to service interactivity and provided by the receiver. Those resources/information may comprise application launch pages and, for example, corresponding Javascript, CSS and XML documents which provide the logic, mark-up and display control of the broadcaster application and/or media files for rendering under application control. Such resources may be delivered to the receiver via broadcast, or fetched from a network server by the application via broadband/Internet access. If a broadcaster application requires access to broadcast-delivered resources from the receiver, or if the application requires the receiver to perform specific actions not defined by the User Agent APIs, it can request that resource from a built-in WebSocket server in the receiver via a set of ATSC-defined JSON-RPC messages as specified in A/344 [7].

6.2.2.1 Application Signaling

A/337 [6] defines application signaling in the form of an XML-based Service Layer Signaling metadata fragment called HELD (HTML Entry pages Location Description). This metadata fragment specifies the properties and other information of files belonging to broadcaster application(s) associated with ATSC 3.0 linear service(s). Those properties/information include:

- the delivery method (broadcast, broadband or via both transports) of application files,
- whether a file associated with the file URI ('Content-Location' of the extended FDT describing that file) is an individual launch page for the application, or a package of files (aggregated using multipart MIME) which includes the launch page,
- the TSI value of the LCT session/channel for broadcast delivery of application files using the ROUTE (Real-time Object delivery over Unidirectional Transport) protocol as defined in the ATSC A/331 standard [8],
- time intervals during which application content such as media files are broadcast prior to their use by the application.

The XML schema of the HELD is shown in figure 6.1.

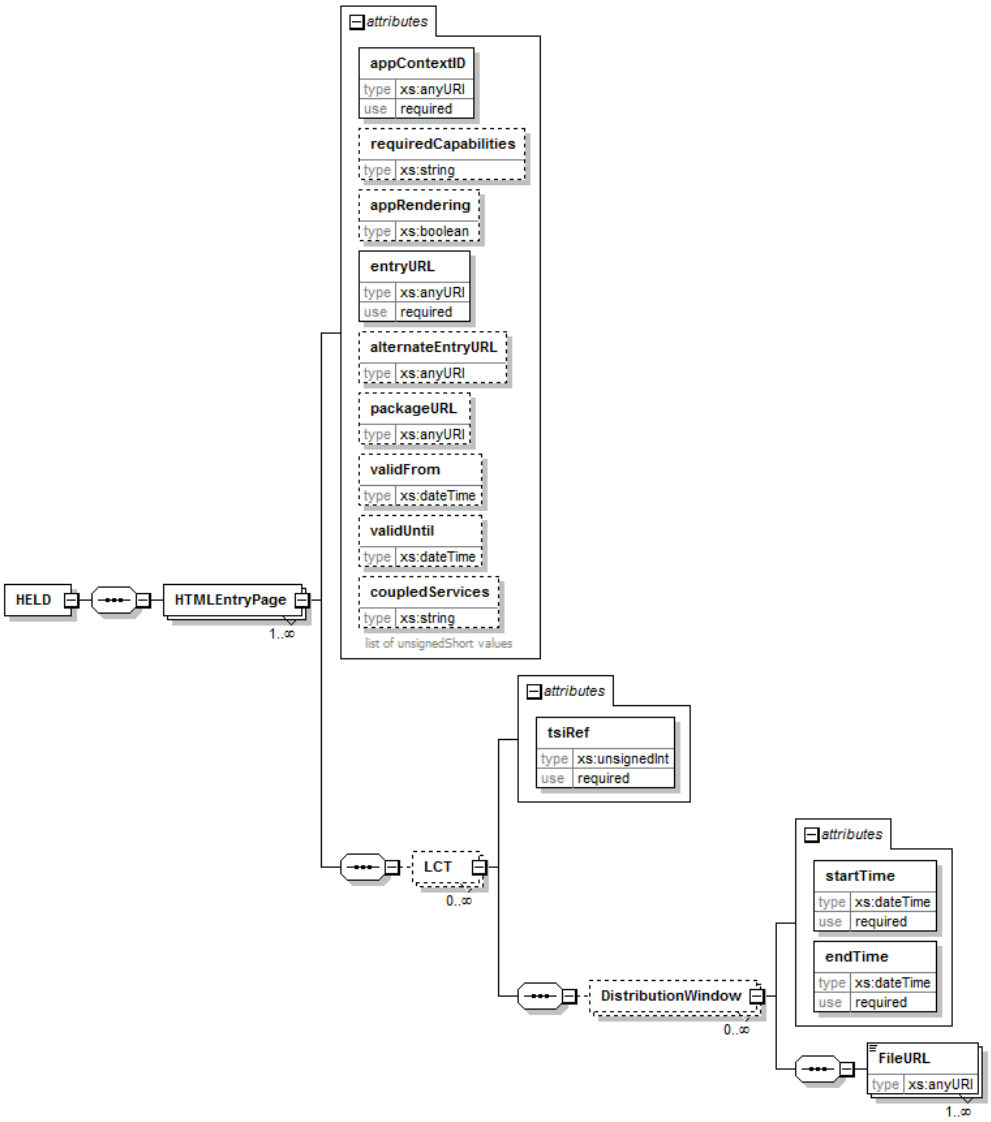

Figure 6.1: XML Schema of the HELD

Notifications of actions to be taken by broadcaster applications, or Events, may be delivered by broadcast or broadband. For a broadcast streaming service associated with a broadcast application and delivered by the ROUTE protocol (i.e., corresponding to a DASH-formatted streaming service), the DASH Event mechanism is used for the notification delivery. DASH Events may be delivered using either of the two mechanisms for Event delivery as defined in the MPEG DASH specification [9]:

- via `EventStream` element(s) appearing in a **`Period`** element of the MPD;
- as Event(s) carried in '`emsg`' box(es) appearing in Media Segments, with their presence signaled by one or more `InbandEventStream` elements of the **`Representation`** element in the MPD.

These two delivery mechanisms may be mixed in the carriage of an Event. A single Event stream may include some Events delivered via an **`EventStream`** element and others delivered via `emsg` boxes.

When delivered via broadcast in an MMT-based system, Events may be delivered in an XML document called an Application Event Information (AEI) document, whose syntax and semantics are very similar to that of DASH Events.

6.2.2.2 Interactivity Content

From the service interactivity perspective, A/344 [7] defines the interactions between one or more service interactivity specific broadcaster applications, running in the ATSC 3.0 receiver, and the receiver platform/middleware. As indicated in clause 6.2.2, A/344 specifies a W3C-compliant User Agent/browser environment within which broadcaster applications are run. The application provides the user with enhanced functionality linked to the main service or programs, such as interactivity features. An interactivity application may include additional assets such as JavaScript files, images and multimedia files which provide, for example, user interface and interactivity display control functions. If the interactivity application requires access to broadcast-delivered resources from the receiver, or if the application requires the receiver to perform specific actions not defined by the User Agent APIs, it can request that resource from a built-in WebSocket server in the receiver, via a set of ATSC-defined JSON-RPC messages. These JSON-RPC messages allow the application to a) query information that was gathered or collected in the receiver, b) receive notifications via broadcast signaling, and c) request the receiver to perform actions that are not otherwise available via the standard JavaScript APIs. As described in clause 6.2.2.1, the received notifications may comprise DASH Event messages which trigger certain UI functionality or media assets to be presented to the user at specific times during the playout of the main program.

Figure 6.2 below depicts a logical representation of the functional components in the ATSC 3.0 receiver.

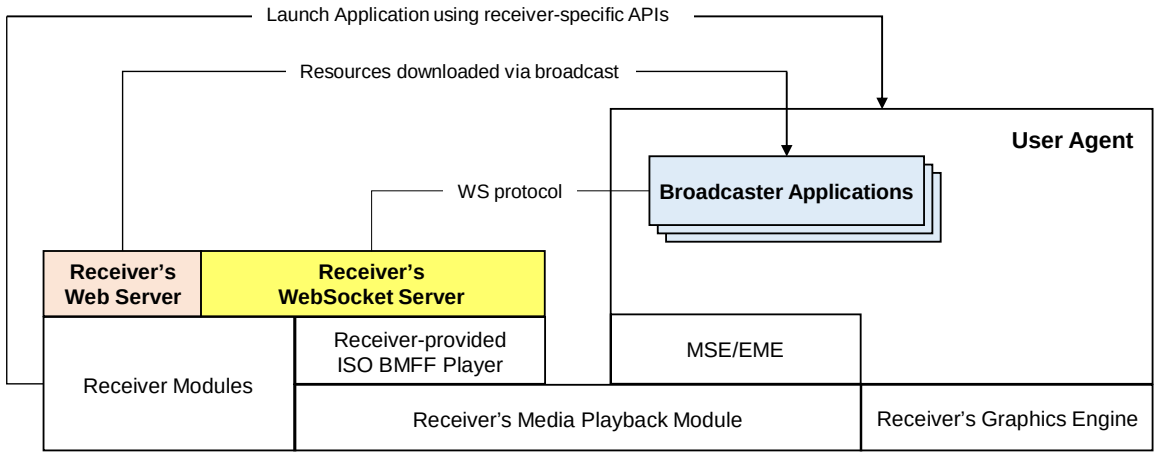

Figure 6.2: Logical Architecture of ATSC 3.0 Receiver

The interactivity application is launched after the receiver obtains application signaling information (as described in clause 6.2.2.1) and forwards the launch URL to the User Agent, which, in turn, loads the entry application document from the URL. Once the interactive application is running, it may request content from various local or external URLs. Any content received over broadcast via ROUTE file delivery will be saved into a local receiver cache associated with and accessed from the receiver's local web server.

7 Interactivity Support for Streaming and Download Services

7.1 Component Model for Interactivity

Figure 7.1 depicts the proposed component model for service interactivity in the context of live streaming service delivery. It is not a network architecture, but represents how interactivity signaling and data components as shown enable the launch and execution of the interactivity service logic in providing the interactivity experience to end users. The model is applicable to either unicast or broadcast/MBMS transport mode for the service.

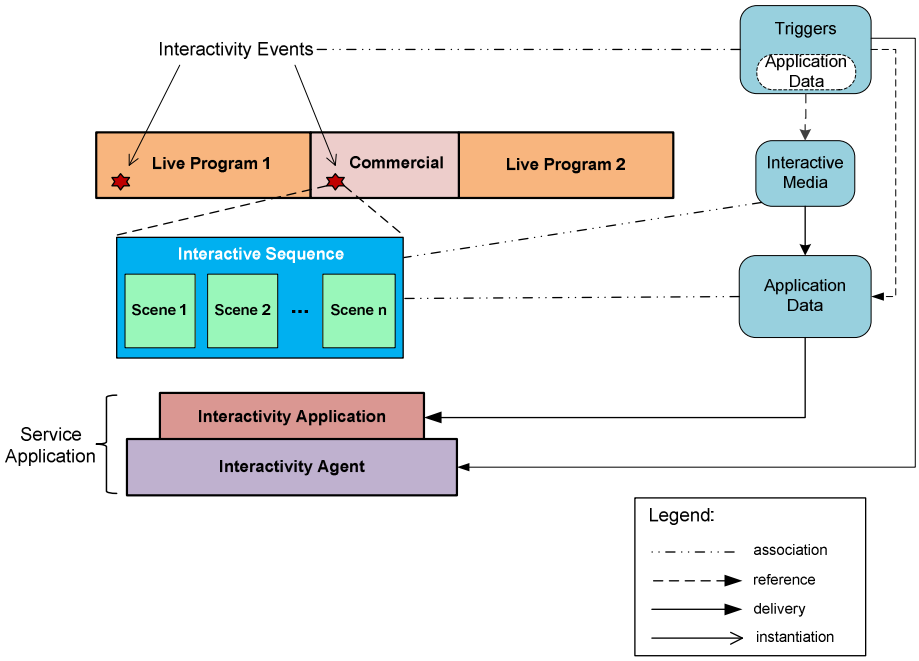

Figure 7.1: Component Model for 3GPP Service Interactivity – Live Streaming Service Delivery

In this component model, the event notification mechanism provides dynamic indication of the occurrence of interactivity events. Event notifications may contain or reference application data, as well as references interactivity media that are used and displayed by the interactivity application which is executed by the interactivity agent. The execution of the interactivity application results in the scene display of scenes associated with the interactivity sequence.

These building blocks of the component model are further described as follows.

- *Event Notifications.* Event notifications provide the dynamic indications of interactivity events and their timing during a content segment (main content or advertisement) to an interactivity-aware service application. The presentation of interactivity events to the end user involves the display of interactivity-specific media content, and may include explicit user engagement with that content. The event notification may be tied to a non-deterministic and real-time event such as a time-out called during a live football game. A potential event notification mechanism for interactivity event notification is DASH Events as defined in MPEG DASH standard [9].

- *Interactivity Media.* These correspond to interactivity media content, such as video clips, images or text files, to be played out during the interactivity occurrences launched by event notifications.

- *Application Data.* Application data pertains to the description of scene information for an interactive sequence. It may include display icons, layout information, or the text of display buttons or announcements to be overlaid on, or presented in line with, the interactivity media. Application data is referenced by, or could be directly carried in, the event notifications (e.g. in the usage of DASH Events, corresponding to the **`Event`**`@messageData` attribute, or the `message_data` field of '`emsg`' box).

- *Interactivity Sequence.* This comprises a set of one or more interactivity scenes to be rendered during an interactivity event. For example, an ad which includes sidebar display of a hyperlink for additional information on a sports car, and which when clicked by the user, may lead to an offer for entering sweepstakes drawing to win that car. The contents making up those scenes may consist of a combination of interactivity media s as described above, and application data.

- *Interactivity Application.* An interactivity application is a component of the service application that contains the logic associated with a specific interactivity use case (i.e. separate logic pertaining to "Click for Info" vs. "Voting" use cases). It may be delivered as a file to the device to be stored in advance of the interactivity event occurrence.

- *Interactivity Agent.* The interactivity agent is another component of the service application. It executes the interactive application logic and provides rendering capabilities of scenes to be displayed during an interactivity sequence. It may also implement a pre-determined overall layout for a given interactivity use case, for example, indicating where on the screen the interactivity content will be displayed relative to the main program. In one possible implementation, the Interactivity Application takes the form of a Web application (HTML/Javascript) and the Interactivity Agent is the web runtime engine required for interpreting and executing the app.

7.2 Interactivity Event Notification Functions

7.2.0 General

The main functionality of the Interactivity event notification mechanism are described in clause 7.1, and in the case of DASH service delivery, in the interactive service architecture and related message flows indicated in clause as well in clause 5.1 of the present document. In addition, interactivity notification functionality is described in clauses 6.1 and 6.2 for DVB and ATSC 2.0 interactive applications/services, respectively.

Interactivity event notifications provide dynamic indications of the occurrence of interactivity events, and related timing information. The event notification may be tied to a non-deterministic and real-time event such as a time-out called during a live football game. Interactivity event notifications may contain or reference application data, as well as references interactivity media that are used and displayed by the interactivity application. Interactivity event notifications and their associated message contents are delivered to the UE, and are typically forwarded to and processed by the responsible application entity, for example the Interactivity Agent as described in clause 7.1 and as shown in figure 7.1. In the case of DASH services delivered over the unicast or MBMS bearers, and assuming the use of DASH Events to convey the interactivity event notifications, DASH Events, in the form of an Event Stream, are produced by the DASH Segmenter/MD Generator. The DASH Event Stream is then sent to the DASH client by the HTTP Server or BM-SC, via unicast or broadcast service delivery, respectively, to be in turn forwarded to the Interactivity Agent as shown in figures 5.1 and 5.2.

7.2.1 DASH Events

A potential event notification mechanism for interactivity event notification is DASH Events as defined in MPEG DASH standard [9], and referenced in the Rel-13 3GP-DASH specification, TS 26.247 [10]. DASH Events are logically generic notification messages that can be contained either in the MPD or inband to the Representation, to signal aperiodic information to the DASH client or to an application. More details on the use of DASH Events as interactivity event mechanism are provided in clause 9.1.1.

7.2.2 HTML5 Text Track

This interactivity event mechanism is specific to the use of Web applications as the service application and/or interactivity agent which executes the interactive application logic pertaining to the scenes to be displayed during an interactivity event. It might not be applicable to a service application/interactivity agent implemented as a native application, i.e., written for a certain mobile device or platform. As defined in the W3C HTML5 standard [11], a media element (e.g. audio, video) can have a group of associated *text tracks*, known as the media element's list of text tracks. The text tracks corresponding to the <track> element children of the media element. The <track> element and its associated text tracks provide a standardized mechanism to add subtitles, captions, screen reader descriptions, chapters and metadata to video and audio, via the attribute `kind`, which can be `subtitles`, `captions`, `descriptions`, `chapters` or `metadata`. Each text track has a corresponding `TextTrack` object. The track element's `src` attribute points to a text file that holds data for timed track *cues*, which can potentially be in any format a browser can parse.

The ability to carry structured data in cues enables flexible use of the track element. For example, for interactivity support, the type (i.e., `kind`) of cue data may be set to `metadata`, for use by Javascript. Such cue data could convey interactivity notifications to which an interactivity-enabled Web application can listen for as cue events, extract the text of each cue as it fires, parse the data, and then use the results to make DOM changes for presentation of the interactivity display, synchronized with media playback. One the other hand, when cue data is not of a form such as captioning or subtitles, which can be handled directly by the media player, the application is then responsible for handling it. There can be issues, however, in the timing of how the application receives this event data, which has been identified by the HbbTV Association [12]. In particular, interactivity events of duration less than 250 msec may be missed by the application.

8 Summary of Working Assumptions, Recommended Requirements and Potential Solution for Service Interactivity at the Application, Service and Transport Levels

8.1 Application-Level Assumptions, Needs and Solutions

The working assumptions, recommended requirements, and potential solution frameworks at the application level for supporting service interactivity are as follows:

- Interactivity applications types to be supported include mobile TV with auxiliary data and user interactivity, click-for-information, broadcast delivery of live events with dynamic and interactive ad insertion which can be personalized.
- It ought to be possible for an interactivity application, whose logic is executed by the interactivity agent, to be implemented as any of the following types:
  - a *native application*, written for a certain mobile device or platform,
  - a *web application*, written in HTML/Javascript/CSS, downloaded from a web site, and runs in the device's web browser, or
  - a *hybrid application* in the form of a web app wrapped inside a native container which provides access to native platform features
- Interactivity application related content can be distinguished among the following types: application software/code, scene update information, and media asset files
- For live DASH services and whereby DASH Events carry interactivity event notifications, the DASH client will expose an API to the interactivity-enabled application via the interactivity agent to register for callbacks, in order to obtain interactivity event notifications.
- The DASH client could be implemented as middleware in the multimedia subsystem of the device's OS, or included as a component of the interactivity application logic.
- It is possible to use the DASH Event messages to provide interactivity related information including start time of the interactivity event occurrence and its duration, and application-specific data associated with that interactivity event.
- For a web-based interactivity application, the interactivity event notification mechanism could be implemented as DASH Events, as timed track cues carried in the HTML5 <*track*> element, or via the use of WebSockets for server-push of event notifications to the client application.
- There ought to be a means for the interactivity application to offer a personalized interactivity user experience to the end user, for example, via access to user profile or preference information contained in the device or obtained from the network.

8.2 Service-Level Assumptions, Needs and Solutions

The working assumptions, recommended requirements, and potential solution frameworks at the service level for supporting service interactivity are as follows:

- Service signaling, for example via the MBMS USD, can uniquely identify an interactivity application data file, containing interactivity application logic, as one of the content components of an interactivity-enabled User Service and associated with a specific interactivity use case, from other content files associated with interactivity support, such as media assets such as video clips, images or text files, to be played out during the interactivity event.
- Service signaling will indicate the transport mode, and in the case of broadcast delivery, the delivery schedule of interactivity application content files.
- Content intended for program-synchronized interactive display can be cached in the UE for subsequent rendering, at specific times associated with interactivity events, alongside with or in replacement of the main program.
- It is possible to utilize MBMS User Service Discovery/Announcement to provide a notification mechanism whose occurrences are well-known/fixed or unpredictable/dynamic, to inform a service interactivity application in the UE about impending interactivity event occurrences.
- Depending on whether the interactivity application is native or web-based, the interactivity event notification mechanism may be implemented in different ways – for example, as DASH Events carried inband to Representations or out-of-band in the MPD, as timed track cues carried in the HTML <*track*> element, or via WebSocket delivery from a local or network-based server to the client application.
- The service signaling mechanism can support customized/personalized interactivity experiences to be provided among users in a pure broadcast service/content delivery context, i.e., without requiring unicast transactions between the interactivity application and a network server.

It is possible to employ a dedicated MBMS User Service for delivering common files shared by one or more interactivity applications associated with multiple MBMS User Services. As an example, all live football games broadcasted by an MBMS service provider might use same template for interactivity events.

8.3 Transport-Level Assumptions, Needs and Solutions

The working assumptions, recommended requirements, and potential solution frameworks at the Transport level for supporting service interactivity are as follows.

- Content intended for program-synchronized interactive display can be delivered to the UE for subsequent rendering, at specific times associated with interactivity events, alongside or in replacement of the main program.
- In the case of live DASH services and depending on the interactivity application type (native or web based), interactivity event notifications can be:
  - Carried as DASH Events, either inband to Representations by Event Message boxes ('emsg'), or inside the MPD;
  - Transmitted as timed track cues in the HTML5 <*track*> element;
  - Delivered via WebSockets-based server push
  - Sent via broadcast or unicast delivery

9 Utilization of Existing Tools in MBMS and DASH

9.1 Interactivity Event Notification Mechanism

9.1.1 DASH Events

As discussed previously in clause 7.2.1, DASH Events as defined in the MPEG DASH standard [9], and referenced by TS 26.247 [10]. The characteristics of DASH Events are as follows:

- Events are timed – the validity of an Event is defined by a specific media presentation time, and each Event typically has a duration.
- An Event pertains to one of two types of notifications: 1) DASH-specific, or 2) application-specific. In the latter category, an appropriate scheme identifier is used to reference the application to which the DASH client will forward the Event.
- Events of the same type are clustered in Event streams, i.e., a sequence of Event messages of the same type. A DASH client may subscribe to an Event Stream of interest and ignore all other, non-relevant Event Streams.
- Each Event message with the Event stream may contain a message body, whose syntax and semantics is defined by the owner of the scheme identified by the scheme identifier.
- Three types of DASH-specific Events are defined in the MPEG DASH standard [9]:
  - An Event message conveying the impending expiration of the current MPD;
  - An Event message conveying the impending expiration of the current MPD, and in addition, the Event message includes an MPD *Patch*, which complies to the XML Patch Operations framework as defined in IETF RFC 5261 [13];
  - An Event message conveying the impending expiration of the current MPD, and in addition, the Events message encapsulates a complete (and valid) instance of an MPD that updates the to-be-expired MPD

For application-specific Events (i.e., non DASH-specific Events as described immediately above, the MPEG DASH standard [3] does not define the usage of Events. Instead the related semantics and syntax are left to the owner of the Event scheme and associated application, in the form of a `@schemeIdUri` attribute that provides a URI to identify the Event scheme and an optional attribute `@value` defining the value space of that scheme. Such usage of `@schemeIdUri` and `@value` is in accordance to the use of MPD descriptors.

The use of MPD Events (Events signaled in the MPD) is indicated by the presence of one or more **`Period.EventStream`** elements, each instance denoting Events of a common type. The use of inband Events is signaled in the MPD by one or more **`InbandEventStream`** elements in the **`AdaptationSet`** or **`Representation`** element of the MPD. The functionality of an application-specific Event message is the same regardless of whether it's carried in the MPD or inband to a Representation.

The semantics of the **`Period.EventStream`** element is shown in figure 9.1 below.

Element or Attribute Name	Use	Description
EventStream		Specifies an Event Stream
@xlink:href	O	Specifies a reference to an external **EventStream** element
@xlink:actuate	OD default: onRequest	Specifies the processing instructions, which can be either "onLoad" or "onRequest". This attribute shall not be present if the @xlink:href attribute is not present.
@schemeIdUri	M	Identifies the message scheme. The string may use URN or URL syntax. When a URL is used, it is recommended to also contain a month-date in the form mmyyyy; the assignment of the URL must have been authorized by the owner of the domain name in that URL on or very close to that date. A URL may resolve to an Internet location, and a location that does resolve may store a specification of the message scheme.
@value	O	Specifies the value for the event stream element. The value space and semantics must be defined by the owners of the scheme identified in the @schemeIdUri attribute.
@timescale	O	Specifies the timescale in units per seconds to be used for the derivation of different real-time duration values in the **Event** elements. If not present on any level, it shall be set to 1.
Event	0 ... N	Specifies an event and contains the message of the event, formatted as a string. The content of this element depends on the event scheme.
@presentationTime	OD default: 0	Specifies the presentation time of the event relative to the start of the Period. The value of the presentation time in seconds is the value of this attribute divided by the value of the @timescale attribute. If not present, the value of the presentation time is 0.
@duration	O	Specifies the presentation duration of the event. The value of the duration in seconds is the value of this attribute divided by the value of the @timescale attribute. If not present, the value of the duration is unknown.
@id	O	Specifies an identifier for this instance of the event. Events with equivalent content and attribute values in the **Event** element shall have the same value for this attribute. The scope of the @id for each Event is by the @schemeIdURI and @value pair.
@messageData	O	Specifies the value for the **EventStream** element. The value space and semantics must be defined by the owners of the scheme identified in the @schemeIdUri attribute. NOTE: this attribute is an alternative to specifying a complete XML element(s) in the Event. It is useful when an event leans itself to a compact string representation.

Figure 9.1: Semantics of the EventStream element in the MPD

9.1.1.1 DASH Events for Interactivity Notifications

DASH Events, either delivered in the MPD or inband to Segment, fulfil the majority of the necessary functions for initiating the occurrences of service interactivity. A key salient feature of DASH Events, given its definition at the DASH/ISOBMFF level, is that it can be used as the interactivity event notification mechanism for both interactivity-enabled Web applications and native applications. Each Event conveys the start time of the associated interactivity event (as presentation time of the event relative to the start of the Period) and may optionally indicate the validity interval of the interactivity event. In addition, the payload of the Event message, via either the Period.EventStream.Event@messageData attribute of MPD Events, or the message_data [ ] field of the Event Message box 'emsg' for inband Events, can convey the necessary data related to the interactivity event and application. For example, the message may include an Event identifier, application data pertaining to the layout of the interactive display, location where the interactivity media assets to be rendered during the interactivity event can be obtained, information on retiming of the occurrence of the interactivity event, etc.

Another potentially salient feature of the DASH Events mechanism for use as interactivity event notifications is its basic built-in support for personalization of the interactivity experience for the end user. Similar to the @xlink:href attribute in the **Period** element, the **EventStream** element in the MPD may optionally include @xlink:href. As described in the DASH-IF Interoperability Points guidelines [14], as well as in TR 26.848 [15] on targeted advertising functionality, there can be various ways to provide the DASH client a customized remote element entity via XLink [16] resolution. In this case, similar to returning a customized remote **Period** element pertaining to a personalized Ad Period, a customized external **EventStream** element may be returned by the XLink resolver, pertaining to personalized interactivity event notification messages. Different collections of interactivity event notifications whose components are timed to fire at different times, and which may reference different interactivity application or media data, can result in customized interactivity experiences depending on the targeted end users. More recently, the MPEG DASH is undergoing amendments to add support for advanced and generalized HTTP feedback information". Here, the changes to the DASH spec pertain to client-side insertion of custom parameters into HTTP GET requests, to enable customized content to be returned to the HTTP response. In particular, the use of query template in the XLink URL is specified.

The suitability of the DASH Events mechanism as interactivity event notifications may depend on how close in time the interactivity event messages will match the corresponding program incident that initiated the delivery of the Event. The interactivity event signaled by an instance of the DASH Event message is defined to start at a specific media presentation time (relative to the start of the containing Period). For example, should the media content be played out from the time-shift buffer, then the execution of the interactivity event will be delayed by the amount of time shift in the actual play-back of content. Therefore, DASH Events may be unsuitable as interactivity event notifications for launching interactivity features associated with emergency alerts, for which the related interactivity experience will be rendered very close in real time to the occurrence of the alert. On the other hand, for live programs such as a football game or a car race, the program content is typically consumed at the live edge, allowing the time-shift buffer depth to be set to a small enough value such that media time and the real-time are sufficiently close to each another to meet the requirement of the service/content provider delivering the interactivity experience.

10 Summary of Functional Gaps in MBMS and PSS Service Layer Specifications on Interactivity Support

10.1 Introduction

Based on the analysis in clause 4.2.5, there are three areas where MBMS and PSS service layer functionality do not fully support the recommended requirements for service interactivity. These are:

- Notification of Interactivity Incidences;
- Personalization of Interactive User Experience;
- Differentiating Contents in Bundled Delivery of Application Content.

These are individually described in sub-clauses 10.2, 10.3 and 10.4, and a summary is given in sub-clause 10.5.

10.2 Notification of Interactivity Occurrence

In 3GP-DASH, the DASH Event Stream mechanism as defined in TS 26.247 [10], and which in turn references ISO/IEC 23009-1 [9], can serve as the means for providing notification messages, in either a static/pre-defined or dynamic/non-predictable manner, to an interactivity-enabled PSS application to perform application-specific interactivity tasks at specific times. For MBMS, there is no generic definition in TS 26.346 [18] of a notification mechanism to cause interactivity-enabled MBMS applications to perform application-specific interactivity tasks at specific times. However, for DASH-over-MBMS services, it might be possible to use the same DASH Events defined for PSS in TS 26.247 [10] as the interactivity notification mechanism. However, in neither PSS nor MBMS is there a defined API or protocol interface between the DASH client and service application, for the configuration of the DASH client and the service application residing in separate software modules or physical devices. Such API/protocol interface is necessary to enable an interactivity-enabled application or its user agent to asynchronously obtain interactivity event notifications. For example, such API may require the application to register with the DASH client for callbacks, in order to obtain scheme-specific Event streams pertaining to interactivity notification messages. An alternative implementation of such interface may be via the use of a WebSocket connection between the two entities, whereby the DASH client is able to push interactivity-specific Event Stream messages to the application as they are received by the DASH client.

10.3 Personalization of Service Interactivity

There is a use case and derived recommended requirement for providing a personalized interactivity user interface and experience associated with an inserted ad, which itself may either be personalized for individual users, or is generic/common to all recipients. In principle, information about the user, such as a profile or preference list, can be used to enable a personalized interactivity experience. TS 26.346 [18] specifies certain capabilities for targeted content reception, such as by the user's location or group affiliation. However, it's unclear whether and how such general-purpose targeting or personalization of broadcast content reception capability can lead to or enable the occurrence of personalized interactivity events. On the other hand, it ought to be further studied whether MBMS or PSS service layer mechanisms might be defined to enable personalized interactivity in the strictly one-way, broadcast service delivery context. For example, in the case of broadcast DASH, it might be worth considering whether the interactivity notification mechanism conveyed by DASH Event messages can be further leveraged to support personalization of the interactivity occurrences. For example, the Events signaled in the MPD, i.e. the `EventStream` element may contain the attribute `@xlink:href` for obtaining, upon XLink resolution, the Event Stream messages from a remote source. The XLink resolution process could conceivably be designed to enable, for example, via the use of templating and parameter substitution, to allow customized, remote Event messages to be returned based on the end user or user class identification.

10.4 Differentiating Interactivity Content Types

The interactivity application, for example a Javascript document for a Web app-based interactivity application, may be bundled for delivery along with other contents associated with the interactivity application, such as media files to be rendered during the interactivity event. There ought to be a means to uniquely identify the interactivity app from other interactivity content items, so that it can be launched in the UE upon reception, to in turn execute the interactivity tasks for which it was designed, including the acquisition and rendering of interactivity media files at specific times. In NRT file delivery, although multipart MIME is defined as the method for bundling related files, there are no explicitly defined rules on the means to identify the interactivity application among the bundled content items.

10.5 Measurement and Reporting of Interactivity Usage

Details of the interactivity usage measurement and reporting can be found in clause 4.2.5.1.5. In summary, the main deficits in existing MBMS and PSS specification for interactivity support pertain to the following aspects:

- *Signaling*. Additional signaling will need to be defined to enable the service/content provider to inform the UE of the specific parameters (and associated syntax) to be collected that pertain to interactivity usage, when that information ought to be reported by the device to the network, and the protocol/mechanism associated with such reporting. Such signaling will enable the network to indicate and control the reporting such as by percentage of devices to send reports, designation of random sampling or by targeted devices to provide interactivity consumption reports, and the times to perform the reporting.

- *Opt-in.* There ought to be a means for the service provider's intention to collect a user's interactive engagement with the main service/program to be made known to the user, and allow explicit user opt-in to such data collection and reporting.

- *Secure storage.* A means ought to be defined to ensure that interactivity usage data measured by the UE will be securely stored on the device, and in the network.

10.6 Gap Analysis Summary

In summary, the following potential gaps in the MBMS and PSS service layer specifications are identified:

- There is need for a notification mechanism to signal the impending occurrence of either a scheduled or unscheduled event upon which time an interactivity experience could be provided to the user.

- In relation to the notification mechanism, a means ought to be devised for an interactivity-aware application to be informed about the impending occurrence of, and obtain relevant information for, a service interactivity event, in order to provide the appropriate interactivity experience to the user during the interactivity event.

- A signaling mechanism ought to be provided to enable customized/personalized interactivity experiences to be delivered to different users, in a pure broadcast service/content delivery context, i.e., without requiring unicast transactions between the interactivity application and a network server.

- A method ought to be defined, in the case of bundled delivery of interactivity-related content items, to identify the interactivity application from interactivity assets contained in the bundle.

11 Application/Presentation vs. Transport/Service Layer Functions on Interactivity

11.1 Application and Presentation Level Functionality for Interactivity Support

Application software are computer programs designed to perform a set of coordinated tasks or activities for the benefit of an end user. Application software based services, or application services, make use of service enabling functions defined by 3GPP specifications (e.g., signaling, transport, QoS management, security, codecs, and logical and physical

channels) to deliver purpose-built functionality to subscribers of those services. Application functionality is largely outside the scope of 3GPP specifications, but are the purview of software developers and providers of application or content services. Similarly, applications which provide the interactivity user experience and user interface are outside the domain of 3GPP application and service layer specifications. However, these interactive applications might be able to leverage functionality offered or referenced by 3GPP specifications such as APIs offered by the UE for access to MBMS User Services, defined as part of the TRAPI work item and in TS 26.347 [19].

The presentation layer contains the components that implement and display the user interface and manage user interaction. This layer includes controls for user input and rendering, and components that organize user interaction, and is an essential functionality in interactive services. The presentation layer typically includes the following functionality:

- *User Interface components*, as the application's visual elements used to display information to the user and accept user input.
- *Presentation Logic components*, comprising the application code that defines the logical behavior and structure of the application in a way that is independent of any specific user interface implementation.

Handling of user input, events, and personalization information can be additional functional components of the presentation logic.

3GPP has several specifications pertaining to the presentation layer that could be used to support interactive applications:

- HTML5, Javascript and CSS and various APIs to support media synchronization, SVG animation, scene updates, etc., for use by web app-based interactive apps, as described in TR 26.907 [20].
- DIMS (Dynamic and Interactive Multimedia Scenes) as defined in TS 26.142 [21] enables display and interactive control of multimedia content functionality that could be used by for native interactive applications.
- SMIL (Synchronized Multimedia Integration Language) based scene description for a multimedia presentation, as defined in TS 26.234 [22], and usable by web app-based interactive apps.

NOTE: References to DIMS and SMIL are for informational purposes, as these specifications are no longer maintained and promoted by 3GPP. The preferred 3GPP presentation layer document is TR 26.907 [20].

11.2 Service Layer Functionality for Interactivity Support

11.2.0 General

Service layer features defined in 3GPP MBMS and PSS specifications and might be considered relevant for interactivity support comprise the following functional areas: transport, signaling, and general "service management" such as security and reporting of interactivity usage.

11.2.1 Transport and Signaling Functions

Interactive applications required for interactivity services could be pre-installed or either carried by a download delivery method via broadcast, or retrieved using the unicast bearer along with associated auxiliary data. Interactivity media could be similarly delivered via broadcast or unicast bearers. The application and associated assets including interactivity media and auxiliary data can be carried together as an aggregate document, for example in a multipart MIME file. When the MBMS download delivery method is used, separate FLUTE sessions, each referenced by an instance of the *deliveryMethod* element, could be used to carry them. Properties of each FLUTE session that carry interactivity-related content are provided by a Session Description instance documents, or SDP file, similar to the use of SDP files to describe the FLUTE sessions that carry the main service contents. The interface between interactive applications and PSS or MBMS clients could enable interactive application to access its associated assets delivered via broadcast or unicast delivery, so that the associated assets can be presented or consumed by the application.

In broadcast delivery of interactivity apps and associated assets, signaling functionality is expected to provide a mechanism to distinguish between the interactivity applications from associated assets. The signaling functionality might include the version of the interactivity app, to enable the UE to determine whether it is capable of supporting the user interface or media presentation capabilities corresponding to that version of the app. Multiple versions of an interactive app and associated media contents could be delivered, to allow the UE to download the version that it can process. The signaling functionality could also indicate when an interactive application is needed to be loaded or launched so that the interactivity app can initiate the interactivity tasks it is designed for and at the appropriate time. The signaling functionality could also indicate when an interactive application needs to be unloaded or terminated so that the interactivity app can be stopped at the appropriate time for reasons such as for displaying an inserted targeted advertisement, or switching to another application. Unloading or termination could be performed by the application's own logic.

Another important signaling function in support of interactivity is to provide notification of impending occurrences of either scheduled or unscheduled events, upon which time an interactivity experience is expected to be provided to the user, as described in clauses 4.2.4.1 and 4.2.5 of the present document.

11.2.2 Service Management Functions

Service management functions in support of interactivity might include the measurement and reporting of interactivity consumption, [as described in clause 12]. In the case of MBMS, associated delivery procedures such as file repair and reception/QoE reporting could be applicable in the delivery of interactivity-related content components, similar to the delivery of content components of the main service with which the interactivity is associated. Another MBMS service management function that might be used in a common way between the delivery of interactivity-related content components and user service components is service protection of download data as described in clause 6.6.3 of TS 33.246 [23].

12 Measurement and Reporting of Interactivity Consumption

The use case and associated discussion and analysis regarding the measurement and reporting of service interactivity are presented in the sections of related sub-clauses under clause 4.2. It indicates the value or importance to the service or content provider supplying service interactivity functionality, typically associated with a main service or program, to know, at a quantitative level, the usage, e.g., the number of click-throughs, views of interactive content, or other forms of user engagement with interactivity features. Such knowledge could be used by the service/content provider to increase the effectiveness of auxiliary contents or services, associated with main programs, offered to end users, possibly in a targeted or personalized manner, as well as potentially increase service revenue, e.g., via additional advertisement, increased cellular data usage, or e-commerce related to interactive services. At the same time, towards the protection of user privacy, it is expected that the usage of the reported interactivity usage is restricted to the MBMS User Service to which the interactivity events pertain, and assumes that the user has explicitly opted in to such interactivity usage collection and reporting. In addition, the 3GPP operator, on behalf of the end user, may wish to ensure that interactivity usage information is maintained in secure storage on the user device, and that transmission of that information to the network is secure. It ought to be possible for the 3GPP service provider to configure or manage parameters of interactivity measurement and reporting. It ought to also be possible for the 3GPP service provider to specify two types of interactivity usage reporting sessions: by randomly-sampling and by specifically-targeted user groups.

From the gap analysis perspective, there is no specification of interactivity-related usage measurement and reporting functionality in the existing MBMS and PSS service layer specifications. Additional signaling functionality will need to be defined in support of interactivity usage measurement and reporting. Such signaling might include specifying the parameters of interactivity events and their usage to be collected by user devices, syntax and semantics of interactivity usage reports to be sent to the network, and metadata to control the reporting by the entirety or a subset of user devices.

13 Summary and Recommendations

This Technical Report presents the findings of the FS_IS3 study item, "Interactivity Support for 3GPP-based Streaming and Download Services". It describes the potential value of service interactivity associated with 3GPP MBMS and PSS services for service/content providers and end users. The report reviews the technical procedures and necessary enablers to support interactivity features and provides an example end-to-end network architectures in the delivery of DASH streaming services, via unicast and broadcast bearers, to which interactivity mechanisms are associated. It also includes overviews of interactive service mechanisms defined in the ATSC 3.0 and DVB digital broadcast TV standards. The document then provides an analysis on the available, necessary and missing functionality in existing MBMS and PSS specifications for supporting service interactivity. The overall discussion on service interactivity is based on use case descriptions and includes the associated working assumptions, recommended requirements and gap analysis.

In the evaluation of functional gaps in MBMS and PSS "service layer" specifications, specifically TS 26.346 [18] and TS 26.247 [10], the following deficits are identified, along with a brief discussion on the potential solution framework:

1) *Notification of interactivity occurrence*. There needs to be defined notification mechanism(s), appropriate for the different types of services provided by MBMS and PSS, to dynamically trigger interactivity-enabled applications to perform the interactivity-specific tasks for which they are designed, at specific times. For broadcast or unicast DASH services, DASH Events as described in clause 9 may represent a suitable solution. However, similar interactivity event notification mechanism ought to be available in the delivery of other service types such as RTP-based streaming or NRT file delivery service.

2) *Personalization of service interactivity*. There needs to be a specified method to enable personalization of the user interface/user experience during an incidence of an interactive event. Such personalization might make use of information regarding the user/associated device, such as user profile/preference information, or his/her current location.

3) *Differentiating interactivity content types*. There needs to be a defined means to uniquely identify an interactivity application from other interactivity-related content items, such as media files for display under application control, during an interactivity event, when these files are bundled for delivery – for example as a multipart MIME aggregate document.

4) *Measurement and reporting of interactivity usage*. A signaling mechanism will need to be defined to enable and manage the device in the collection and reporting of the usage or engagement by the user of interactivity features, including the capability to selectively control the user/device population to perform the reporting. Also pertaining to the measurement and reporting of interactivity usage is the need to define a service framework to ensure the protection of user identity and privacy, which ought to include the secure storage of collected interactivity usage data in the device.

For the above identified functional gaps and potential solution frameworks, it is recommended to investigate extensions to the existing capabilities defined in the 3GPP MBMS and PSS service layer specifications necessary to fulfil the gaps. Such extensions will likely require future, stage 3 work which is beyond the scope of the FS_IS3 study item and of the present document.

Annex A:
Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2016-09	73	SP-160604			Presented to TSG SA#73 for information		1.0.0
2017-03	75	SP-170032			Presented to TSG SA#73 for approval	1.0.0	2.0.0
2017-03	75				Version for Release 14		14.0.0
2018-06	80				Version for Release 15		15.0.0
2020-07	-	-	-	-	Update to Rel-16 version (MCC)	15.0.0	**16.0.0**

History

Document history		
V16.0.0	September 2020	Publication