



Errata 3 to TS 102 796 V1.2.1

2016-06-07

Copyright 2013-6 HbbTV Association

Table of Contents

1	Introduction	6
2	Conventions	6
3	Summary.....	6
4	Changes to TS 102 796 v1.2.1.....	12
4.1	Clause 2 – References	12
4.1.1	OIPF errata to R1 V1.2.....	12
4.1.2	DASH video resolutions.....	12
4.1.3	DASH Technical Corrigendum 1	12
4.1.4	Origins for carousel delivered applications.....	12
4.1.5	TLS profile	12
4.1.6	XML Parsing	13
4.1.7	Mixed content.....	13
4.1.8	Ignore MPD, Adaptation Set or Representation where no supported profile is used.	13
4.2	Clause 4 – Overview	13
4.2.1	Examples of different application types	13
4.3	Clause 6 – Service and application model	13
4.3.1	Missing text in 6.2.2.4	13
4.3.2	Modification to application lifecycle on channel selection.....	14
4.3.3	Remove reference to external application authorisation descriptor	14
4.3.4	Origins for carousel delivered applications.....	16
4.3.5	Clarify behaviour if AIT PID changes.....	16
4.3.6	Reference to table off by one.....	17
4.3.7	Clarify out of resources and memory behaviour	17
4.3.8	Change channel when no access to broadcast resources	17
4.3.9	Suspend and resume access to broadcast resources	17
4.3.10	Background loading of applications.....	17
4.3.11	Destroying applications	17
4.3.12	Ambiguities in time-shift and immediate recording.....	18
4.4	Clause 7 - Formats and Protocols.....	18
4.4.1	HTTP User Agent	18
4.4.2	Editorial correction.....	18
4.4.3	Remove ApplicationSpecificDescriptor	18
4.4.4	Clarification on DASH video resolutions	19
4.4.5	Application types in the AIT	19
4.4.6	Multiple sections for AIT HbbTV subtable.....	19
4.4.7	Support for previous versions of the specification	19
4.4.8	Ignoring unexpected tables.....	19
4.4.9	XML Parsing	20

4.4.10	Transcoding of audio metadata.....	20
4.4.11	Object carousel caching behaviour.....	21
4.4.12	Constructing the Base URL from the AIT.....	21
4.4.13	Signed / trusted apps.....	21
4.4.14	HTTPS for MPEG DASH.....	21
4.5	Clause 8 – Browser Application Environment.....	22
4.5.1	Mismatch in removeStreamEventListener arguments.....	22
4.5.2	StreamEvent event description.....	22
4.5.3	Mandation of stream event listeners.....	22
4.5.4	statusText attribute of XHR with DSMCC.....	22
4.5.5	XML parsing.....	22
4.6	Clause 9 – System Integration.....	22
4.6.1	Clarification of URL example.....	22
4.6.2	Trick modes with adaptive bitrate content.....	23
4.6.3	DASH technical corrigendum.....	23
4.6.4	Ambiguity when stopping broadband video.....	23
4.6.5	HTTPS support for non-adaptive streaming is not required.....	23
4.6.6	DASH audio adaptation sets and language attribute.....	23
4.6.7	Definition of "trickmodes" for DASH content.....	24
4.6.8	CORS and the dvb: protocol.....	24
4.6.9	AVComponent.encoding property.....	24
4.6.10	Seek accuracy.....	24
4.7	Clause 10 – Capabilities.....	26
4.7.1	Video scaling and visibility of decoded video.....	26
4.7.2	Errors in the example XML capabilities.....	26
4.7.3	Location of the DRM information in the XML capabilities.....	27
4.7.4	Sample application no longer provided.....	27
4.7.5	Extensions to A/V object for parental rating errors.....	27
4.7.6	Maximum cookie size.....	27
4.7.7	Availability of Key Events to Applications.....	27
4.7.8	Download management and manageDownload.....	29
4.7.9	XML Parsing.....	29
4.7.10	Graphics plane resolution.....	29
4.7.11	<u>CICAM capabilities</u>	29
4.7.12	KeySet requirements for text input.....	29
4.7.13	Default component selection.....	30
4.8	Clause 11 – Security.....	30
4.8.1	TLS profile.....	30
4.9	Annex A – OIPF DAE Specification Profile.....	31
4.9.1	getSIDescriptors optional argument.....	31

4.9.2	DVB-SI parental rating	32
4.9.3	Definition of programmeID.....	33
4.9.4	Extensions to video/broadcast object for recording and time-shift.....	33
4.9.5	Missing channel.nid property	34
4.9.6	Misspelling of countryId	34
4.9.7	Incorrect section names	34
4.9.8	Incorrect references to DAE Annex B.....	35
4.9.9	Missing window.onload property	35
4.9.10	Explicitly linking EIT to metadata search	36
4.9.11	Mandatory argument to sendDRMMMessage	36
4.9.12	Queue management on type change	36
4.9.13	Audio from memory	36
4.9.14	Remove use of display:none.....	36
4.9.15	Extensions to A/V object for parental rating errors.....	36
4.9.16	ScheduledRecording constants not included	37
4.9.17	Copy/paste error in A.2.4.2	37
4.9.18	Changes to Extensions for playback of selected media components	37
4.9.19	Origins for carousel delivered applications	37
4.9.20	Changes to OIPF section 8.4.2	37
4.9.21	onPlaySpeedChanged - Should always dispatch when play() is called	38
4.9.22	CEA-2014 Tagged Opcodes Replacement	38
4.9.23	Usage of setAttribute("data",value) method of A/V HTMLObject.....	38
4.9.24	CSS display:none and OIPF objects.....	38
4.9.25	MIME type for HTML documents	39
4.9.26	TLS Profile	39
4.9.27	Encoding parental rating for dvb-si.....	39
4.9.28	JavaScript classes being addressed as "window."	40
4.9.29	XML Parsing	40
4.9.30	AVAudioComponent.audioChannels.....	40
4.9.31	<u>AVComponent for stream type 0x04</u>	41
4.9.32	programmeID in the ScheduledRecording class	41
4.9.33	recordings property in oipfRecordingScheduler	41
4.9.34	Modifications to Clause 7.10.1.1 and references to it.....	41
4.9.35	DRM API restriction to 'Trusted' applications.....	42
4.9.36	CI and video blanking by terminal	43
4.9.37	CI parental_control_info message	44
4.9.38	On-going Recordings in the oipfRecordingScheduler object.....	44
4.9.39	onPlayStateChange and playState	44
4.9.40	No defined mimetype for A/V control object when playing back PVR.....	44
4.9.41	Defining "delivery system descriptor" for DVB S2/C2/T2.....	45

4.9.42	Selecting media components in language not supported by terminal	45
4.9.43	Mixed content.....	46
4.9.44	repeatDays property from ScheduledRecording class.....	46
4.9.45	Modifications to A.2.4.1	46
4.9.46	Ambiguities in time-shift and immediate recording.....	47
4.9.47	Trick modes definition is vague for play() calls with non-DASH media.....	56
4.10	Annex B - Support for protected content delivered via broadband	56
4.10.1	Flags=0 is wrong	56
4.10.2	Broken reference in annex B	56
4.11	Annex E – Profiles of MPEG DASH.....	57
4.11.1	Dynamic MPD served with cache control off	57
4.11.2	Missing text in E.4.2.1.....	57
4.11.3	Capitalisation of roles incorrect	57
4.11.4	DASH video resolutions.....	57
4.11.5	Segment length	57
4.11.6	Switching between representations	57
4.11.7	URI for EAC-3 AudioChannelConfiguration.....	58
4.11.8	XML Parsing	58
4.11.9	Ignore MPD, Adaptation Set or Representation where no supported profile is used.....	58

1 Introduction

This document contains the currently identified and resolved errata to ETSI TS 102 796 v1.2.1. It is a living document which will be updated based on experience of implementing receivers, services and tests. Versions of this document will periodically be made publicly available via the www.hbbtv.org web site.

Errata 3 includes all errata from errata 1 and errata 2 and obsoletes both those documents.

Feedback is welcome. HbbTV members should provide this through the HbbTV internal issue tracking system. Non-members may provide it by email to info@hbbtv.org.

For avoidance of doubt, the contents of this document have not been reviewed or approved by ETSI.

2 Conventions

In this document, text quoted from other documents or to be added to other documents is indented except where it appears in a table. Fine-grained changes in text from other documents are shown using the underline and strikethrough convention.

3 Summary

The following table summarises the changes included in this document.

Issue #	Short Description	Category	Clauses Impacted	Clause in this document
1317	getSIDescriptors optional argument	Ambiguity	A.1	4.9.1
1356	Error in default xmlCapabilities re trick mode support	Ambiguity	10.2.4	4.7.2
1442, 3543	Clarification of URL example	Editorial	9.2	4.6.1
1443	Dynamic MPD served with cache control off	Ambiguity	E.2.1	4.11.1
1446	Sample application no longer provided	Editorial	10.2.5	4.7.4
1451	Incorrect section names	Editorial	A.1	4.9.7
1461	StreamEvent event description	Ambiguity	8.2.1.2	4.5.2
1462	Mismatch in removeStreamEventListener arguments	Editorial	8.2.1.1	4.5.1
1488	Misspelling of countryId	Editorial	A.1	4.9.6
1492	Missing text in 6.2.2.4	Editorial	6.2.2.4	4.3.1
1493	DVB-SI parental rating	Ambiguity	A.1	4.9.2

Issue #	Short Description	Category	Clauses Impacted	Clause in this document
1525	Video scaling and visibility of decoded video	Editorial	10.2.1	4.7.1
1531	Trick modes with adaptive bitrate content	Editorial	9.4	4.6.1
1531	Clarify support for PlaySpeedChange	Error	A.1	4.9.4
1544	Missing window.onload property	Implementation	A.2.8.1	4.9.9
1554	Errors in the example XML capabilities	Ambiguity	10.2.4	4.7.2
1554	Location of the DRM information in the XML capabilities	Editorial	10.2.4	4.7.3
1569	Missing text in E.4.2.1	Editorial	E.4.2.1	4.11.2
1577	Explicitly linking EIT to metadata search	Ambiguity	A.2.9	4.9.10
1581	Missing channel.nid property	Error	A.1	4.9.5
1615	Definition of programmeID	Ambiguity	A.1	4.9.3
1628	Location of the DRM information in the XML capabilities	Ambiguity	10.2.4	4.7.3
1633	Incorrect references to DAE Annex B	Editorial	A.1	4.9.8
1635	HTTP User Agent	Implementation	7.3.2.4	4.4.1
1640	Editorial correction	Editorial	7.3.2.5	4.4.2
1715	Queue management on type change	Ambiguity	A.2.5	4.9.12
1767	Modification to application lifecycle on channel selection	Ambiguity	6.2.2.2	4.3.2
1784	Mandatory argument to sendDRMMMessage	Error	A.1	4.9.11
1848	Remove reference to external application authorisation descriptor	Editorial	6.2.2.3	4.3.3
1887	Audio from memory	Error	A.1	4.9.13
1894	OIPF errata to R1 V1.2	Ambiguity	2.1	4.1.1
1914	Examples of different application types	Editorial	4.1	4.2.1
2007-1	Reference to table off by one	Editorial	6.2.2.7	4.3.6
2043	Broken reference in annex B	Editorial	B.1	4.10.2
2066	Segment length	Error	E.3.2	4.11.5

Issue #	Short Description	Category	Clauses Impacted	Clause in this document
2067	flags=0 is wrong	Ambiguity	B.2.5.1	4.10.1
2068	Remove use of display:none	Implementation	A.2.4.1, A.2.5, new A.2.14	4.9.14, 4.9.45
2083	Extensions to A/V object for parental rating errors	Error	10.2.6, A.1	4.7.5, 4.9.15
2086	Remove ApplicationSpecificDescriptor	Implementation	7.2.3	4.4.3
2121	DASH technical corrigendum 1	Error	2.1, 9.4	4.1.3, 4.6.3
2130	DASH video resolutions	Implementation	2.1, E.4.2.1	4.1.2, 4.11.4
2163	Capitalisation of roles incorrect	Error	E.2.4	4.11.3
2184	ScheduledRecording constants not included	Ambiguity	A.1	4.9.16
2186	Copy/paste error in A.2.4.2	Ambiguity	A.2.4.2	4.9.17
2262	Ambiguity when stopping broadband video	Ambiguity	9.1.1.1	4.6.4
2273	Switching between Representations	Ambiguity	E.4.2	4.11.6
2282	statusText attribute of XHR with DSMCC	Implementation	8.2.2	4.5.4
2298	is audio channels unselect asynchronous?	Ambiguity	A.1	4.9.18
2309	Origins for carousel delivered applications	Error	6.3, A.2.8.1	4.3.4, 4.9.19, 4.1.4
2313	onPlaySpeedChanged - Should always dispatch when play() is called	-	A.1	4.9.21
2396	Clarification on DASH video resolutions	Ambiguity	7.3.1.3	4.4.4
2399	Use of "may" in OIPF section 8.4.2	Error	A.1	4.9.20
2413	Usage of setAttribute("data",value) method of A/V HTMLObject	Ambiguity	A.2	4.9.23
2417	Application types in the AIT	Error	7.2.3.1	4.4.5
2495	Clarify behaviour if AIT PID changes	Ambiguity	6.2.2.3	4.3.5
2497	Changes to the set of available components in video/broadcast object	Ambiguity	A.1	4.9.18
2535	Contradictory language on EIT-pf and EIT-schedule	Error	A.2.9	4.9.10

Issue #	Short Description	Category	Clauses Impacted	Clause in this document
2554	Mandation of stream event listeners	Ambiguity	8.2.1.1	4.5.3
2680	CEA-2014 tagged opcodes replacement	Implementation	A.1	4.9.22
2745	Support for previous versions of the specification	Ambiguity	7.2.3.1	4.4.7
2943	Maximum cookie size	Ambiguity	10.2.1	4.7.6
2971, 5118	Availability of Key Events to Applications	Other	10.2.2	4.7.7
3126	Origin header for carousel origins must have defined case	Ambiguity	6.3.2	4.3.4
3178	CSS display:none and OIPF objects	Implementation	A.1	4.9.24
3214	Multiple sections for AIT HbbTV subtable	Ambiguity	7.2.3.1	4.4.6
3246	AVAudioComponent.audioChannels	Error	A.2.12	4.9.30
3256	Exiting a broadcast-related application	Ambiguity	6.2.2	4.3.2, 4.3.3
3308	Default component selection	Other	New clause 10.2.8	4.7.13
3350	Clarify out of resources and memory behaviour	Ambiguity	6.2.2.4	4.3.7
3388	Change channel when no access to broadcast resources	Ambiguity	6.2.2.7	4.3.8
3390	AVComponent for stream type 0x04	Error	A.2.13	4.9.31
3442	reinforce that HTTPS support for content streaming is not required	Ambiguity	9.1.1.2 and 9.2	4.6.5
3449	MIME type for HTML documents	Ambiguity	A.2.6	4.9.25
3468	TLS profile	Other	2.1, 11.2, A.1	4.9.26, 4.8.1, 4.1.5
3485	Parental rating threshold value for dvb si when no parental rating set	Ambiguity	A.1	4.9.2
3487, 3489	DASH audio adaptation sets and language attribute	Error	9.4, A.1	4.6.6, 4.9.20
3498	Download management and manageDownload	Error	10.2.1	4.7.8
3504	Ignoring unexpected tables	Other	7.2	4.4.8

Issue #	Short Description	Category	Clauses Impacted	Clause in this document
3508	URI for EAC-3 AudioChannelConfiguration	Other	E.2.5	4.11.7
3523	getScheduledRecordings, origins and FQDNs	Ambiguity	A.2	4.9.34
3524	Errors in remove method	Error	A.2	4.9.34
3525	Recordings property in oipfRecordingScheduler	Ambiguity	A.1	4.9.33
3531	XML Parsing	Other	2.1, 7.2.3.2, 7.2.4, 8.2.1.1, 10.2.4, A.1, A.2.6.2, E.2.1	4.1.6, 4.4.9, 4.5.5, 4.7.9, 4.9.29, 4.11.8
3547	Encoding parental rating for dvb-si	Ambiguity	A.1	4.9.27
3595	Definition of "trickmodes" for DASH content	Ambiguity	9.4	4.6.7
3648	JavaScript classes being addressed as "window."	Ambiguity	A.2.8.1	4.9.28
3704	programmeID in the ScheduledRecording class	Error	A.1	4.9.32
3717	Transcoding of audio metadata	Ambiguity	7.3.1.4	4.4.10
3755	Trick modes definition is vague for play() calls with non-DASH media	Ambiguity	New clauses A.2.4.4 and A.2.5.3	4.9.47
3821	Ambiguities in time-shift and immediate recording	Ambiguity	6.2.2.4, A.1, new clauses A.2.14 and A.2.15	4.3.12, 4.9.4, 4.9.46
4028	Signed / trusted applications	Error	7.2.3.1	4.4.13
4030	Mixed content	Other	New clause A.2.6.7,	4.9.43, 4.1.7
4180	KeySet requirements for text input	Ambiguity	10.2.2	4.7.12
4207	On-going Recordings in the	Other	New clause	4.9.34, 4.9.38

Issue #	Short Description	Category	Clauses Impacted	Clause in this document
	oipfRecordingScheduler object		A.2.13	
4269	CI and video blanking by terminal	Error	New clause A.3	4.9.36
4278	CI parental_control_info message	Ambiguity	New clause A.3	4.9.37
4282	CICAM capabilities	Ambiguity	10.2.4	4.7.11
4330	Seek accuracy	Ambiguity	New clause 9.1.3	4.6.10
4390	Defining "delivery system descriptor" for DVB S2/C2/T2	Ambiguity	New clause A.2.4.4	4.9.41
4391	Suspend and resume access to broadcast resources	Error	6.2.2.7	4.3.9
4392	Object carousel caching behaviour	Ambiguity	7.2.2	4.4.11
4393	Graphics plane resolution	Ambiguity	10.2.1	4.7.10
4406	AVComponent.encoding property	Ambiguity	9.4	4.6.9
4432	Background loading of applications	Ambiguity	6.1	4.3.10
4653	HTTPS for MPEG DASH	Ambiguity	7.3.1.2, 9.1.2, 9.2	4.4.14, 4.6.5
4680	No defined mimetype for A/V control object when playing back PVR	Error	A.1	4.9.40
4683	DRM API restricted to 'Trusted' applications	Error	A.1	4.9.35
4684	Destroying applications	Ambiguity	6.2.2.1	4.3.11
4756	repeatDays property from ScheduledRecording class	Ambiguity	A.1	4.9.44
4804	onPlayStateChange and playState	Other	A.2.5	4.9.39
4806	CORS and the dvb: protocol	Ambiguity	9.2	4.6.8
4935	Ignore MPD, Adaptation Set or Representation where no supported profile is used	Ambiguity	E.4.1, 2.2	4.11.9, 4.1.8
4999	Selecting media components in language not supported by terminal	Ambiguity	A.1	4.9.42
5018	Constructing the Base URL from the AIT	Implementation	7.2.3.1	4.4.12

Issue #	Short Description	Category	Clauses Impacted	Clause in this document
5023	Video/broadcast object - use of API when no video component is present	Error	A.2.4.1	4.9.45

Key to categories

Ambiguity	Feature where different implementations may behave in different ways. This includes under-specified features as well as inconsistencies within the specification.
Editorial	Purely editorial change
Error	Clear technical error in the specification. Cannot be implemented as written.
Implementation	Feature removed, simplified or modified in order to simplify implementation and testing.

4 Changes to TS 102 796 v1.2.1

4.1 Clause 2 – References

4.1.1 OIPF errata to R1 V1.2

Implementations shall take into account any changes to the OIPF specifications as indicated in the OIPF Release 1 IPTV Solution V1.2 Errata 1.

4.1.2 DASH video resolutions

The following additional normative reference shall be added;

- [36] Digital Video Broadcasting (DVB); MPEG-DASH Profile for Transport of ISO BMFF Based DVB Services over IP Based Networks

NOTE: Currently available as DVB Blue Book A168

4.1.3 DASH Technical Corrigendum 1

The following additional normative reference shall be added.

- [37] ISO/IEC 23009 1 (2012): “Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats, TECHNICAL CORRIGENDUM 1”

4.1.4 Origins for carousel delivered applications

The following additional normative reference shall be added.

- [38] W3C Recommendation (16 January 2014) “Cross-Origin Resource Sharing”

4.1.5 TLS profile

The following normative references are added;

[39] IETF RFC 5746 “Transport Layer Security (TLS) Renegotiation Indication Extension”

[40] IETF RFC 6066 “Transport Layer Security (TLS) Extensions: Extension Definitions”

4.1.6 XML Parsing

The following normative references are added;

[41] W3C Recommendation (4 February 2004): "Extensible Markup Language (XML) 1.0 (Third Edition)"

NOTE: Available at <http://www.w3.org/TR/2004/REC-xml-20040204/>

4.1.7 Mixed content

The following informative reference is added;

[i.7] W3C Candidate Recommendation, (08 October 2015): “Mixed Content”

NOTE: Available at <https://www.w3.org/TR/mixed-content/>

4.1.8 Ignore MPD, Adaptation Set or Representation where no supported profile is used

The following informative reference is added;

[i.8] DASH Industry Forum:(V3.2) “Guidelines for Implementation: DASH-IF Interoperability Points”

NOTE: Available from <http://dashif.org/wp-content/uploads/2015/12/DASH-IF-IOP-v3.2.pdf>

4.2 Clause 4 – Overview

4.2.1 Examples of different application types

Modify the first set of bullets as follows:

- Broadcast-independent application (i.e. not associated with any broadcast service). This type of application is downloaded via broadband and accesses all of its associated data via broadband.
 - Examples of this type of service are catch-up services and games where the application does not need access to any broadcast resources.
- Broadcast-related application (i.e. associated with one or more broadcast services or one or more broadcast events within a service) that may be launched automatically ("autostart") or explicitly upon user request. This type of application may be downloaded via broadband or broadcast and may access its data via either method.
 - Examples of this type of service are electronic program guides and teletext-like services where the application may wish to present the broadcast video in a window and access other broadcast resources (e.g. EIT metadata)..

4.3 Clause 6 – Service and application model

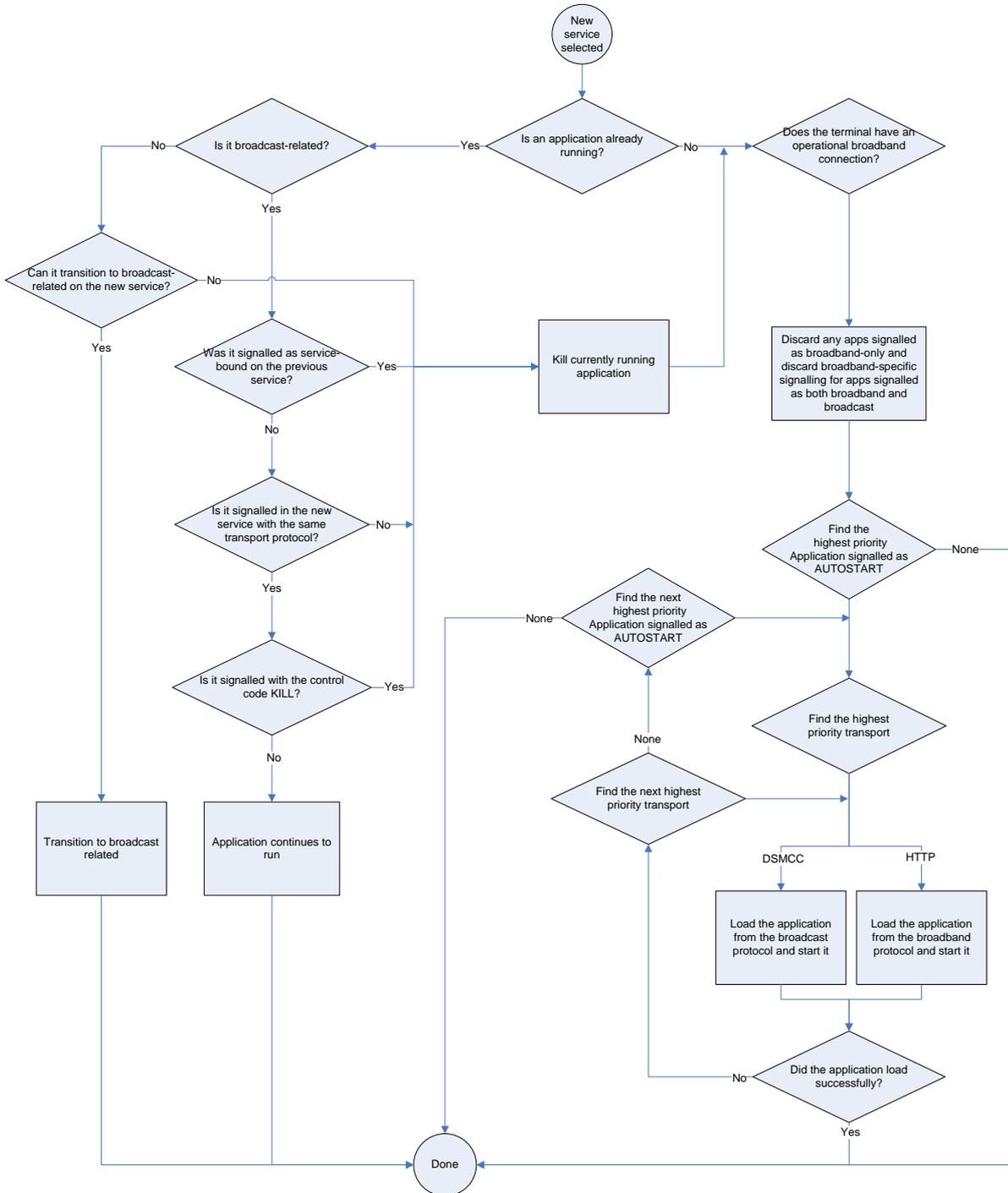
4.3.1 Missing text in 6.2.2.4

Modify the text in section 6.2.2.4 as follows:

- The timeShiftSynchronized property of the Configuration class shall be set to true (see clause A.2.4.3).

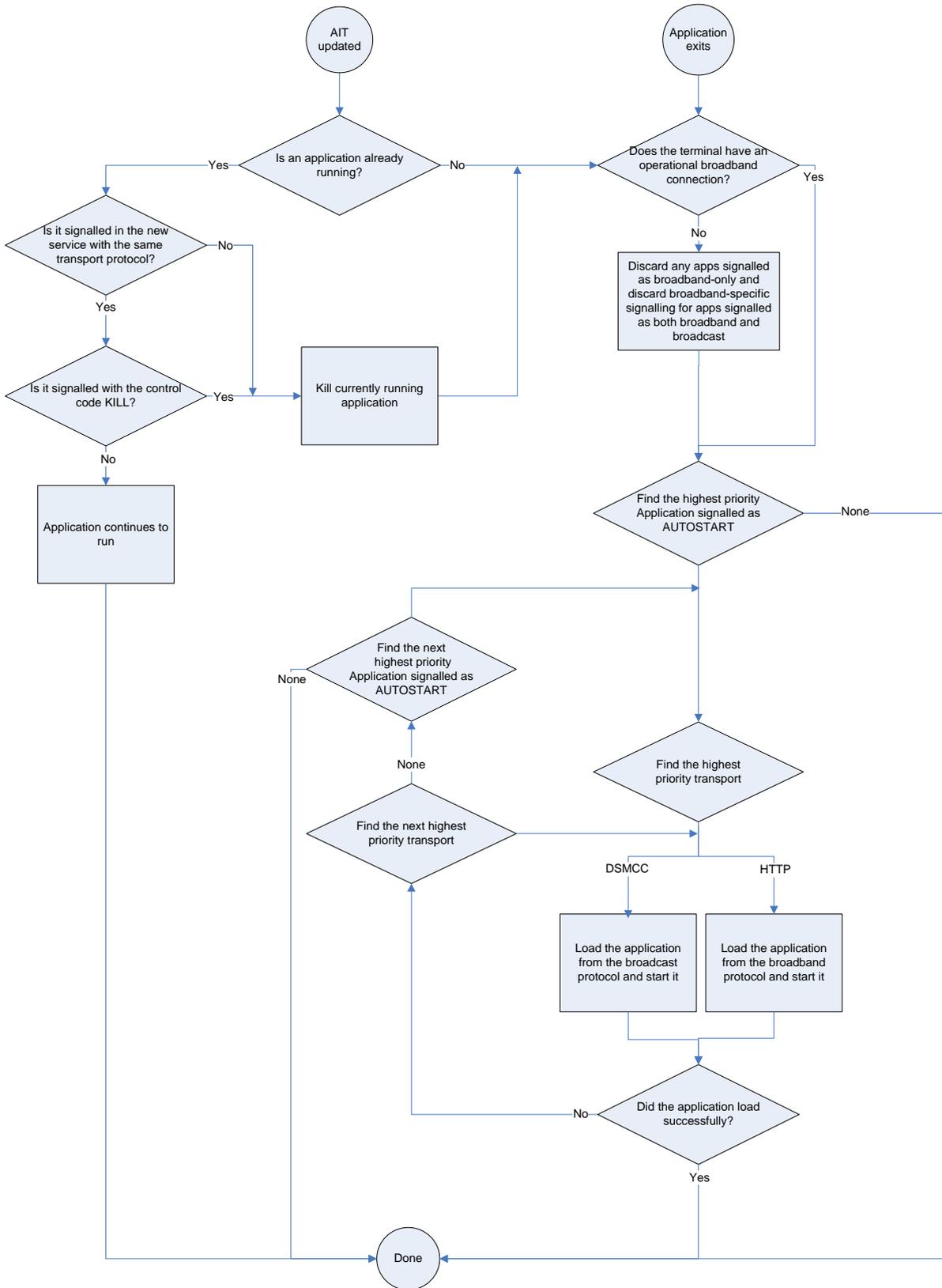
4.3.2 Modification to application lifecycle on channel selection

Replace Figure 13 with:



4.3.3 Remove reference to external application authorisation descriptor

Replace Figure 14 with:



4.3.4 Origins for carousel delivered applications

The following changes shall be made to clause 6.3.

- The existing text shall be renumbered to 6.3.2 “Application boundary”
- The heading for 6.3 itself shall be renamed to “Origins and application boundary”
- A new clause 6.3.1 shall be inserted as below.

6.3.1 Origin

An origin can be derived for each document or other resource loaded by the terminal.

- For resources loaded via HTTP or HTTPS, the origin shall be derived from the URL as defined in the "The Web Origin Concept" specification [1].
- For resources loaded via DSMCC object carousel, the origin shall be the DVB URI in the form (as defined in TS 102 851 [xx] section 6.3.1):

```
"dvb" ":" "/" original_network_id "." transport_stream_id "." service_id "."  
component_tag
```

NOTE 1: In this case, the "host" is the DVB triplet plus the component_tag.

Hexadecimal digits in the DVB triplet and the component_tag shall be encoded using lower case characters.

For resources loaded via DSM-CC, this origin shall be used in all cases where a document or resource origin is used in web specifications including but not limited to cross-origin resource sharing [38].

- “An application boundary is a set of URL origins and object carousels” shall be replaced with
“An application boundary is a set of URL origins as defined in clause 6.3.1”
- “default” shall be added in the following point as shown;
“For applications loaded via object carousel, the default application boundary shall include the carousel from which the first page of the application was loaded”
- The following point shall be deleted;
“For files requested with XMLHttpRequest, the Same-Origin Policy shall be extended using the application boundary i.e. any origin in the application boundary will be considered of same origin”.

4.3.5 Clarify behaviour if AIT PID changes

In clause 6.2.2.3, the following text shall be added immediately following note 3.

The PID on which an AIT component is carried may change. Terminals shall treat this in the same manner defined in clause 5.3.4.2 of ETSI TS 102 809 for the case where an AIT is removed from the PMT and then reinstated. This means that the sub-table shall be considered to have changed, regardless of whether the AIT version number changes, and the normal “AIT updated” sequence defined in Figure 14 shall be followed.

4.3.6 Reference to table off by one

In clause 6.2.2.7, the table in table "State transitions for the video/broadcast embedded object" of the OIPF DAE specification [1] is table 12 and not table 11 as shown.

4.3.7 Clarify out of resources and memory behaviour

In clause 6.2.2.4, the last point in the bulleted list of when an application shall be stopped has "(except as shown below)" inserted as shown.

- The terminal has run out of resources for executing the application (except as described below) and therefore has to terminate it in order to keep operating correctly.

The following text is inserted immediately after the bulleted list.

An application shall not be stopped due to a failure to load an asset (e.g. an image file) or a CSS file due to a lack of memory, although this may result in visual artefacts (e.g. images not being displayed). Failure to load an HTML or JavaScript file due to a lack of memory may cause the application to be terminated.

4.3.8 Change channel when no access to broadcast resources

The following text is added to the end of clause 6.2.2.7.

Access to broadcast resources shall be automatically restored if a channel change is made either by the application or by the user (e.g. by pressing P+ or P-). This may result in the presentation of broadband delivered video being halted. If presentation is halted then this shall be reported using error 3 in the case of the AV Control object or MEDIA_ERR_DECODE in the case of an HTML5 media element.

4.3.9 Suspend and resume access to broadcast resources

In clause 6.2.2.7, the following text is amended as shown;

For a video/broadcast object in the presenting state, Ssuspension of access to broadcast resources shall be treated as a transient error as defined in table 8 of the OIPF DAE specification [1]. The PlayStateChange event that is dispatched shall have the error code 11. For a video/broadcast object in the stopped state, no state changes shall occur and no events shall be generated.

When access to broadcast resources is restored following earlier suspension of access, for a video/broadcast object that was in the presenting state, this shall be treated as recovery from a transient error as defined in table 8 of the OIPF DAE specification [1].

4.3.10 Background loading of applications

In clause 6.1, the following text is removed.

It is optional for a terminal to support background preloading and rendering of applications other than the visible one.

4.3.11 Destroying applications

The following changes are made to clause 6.2.2.1.

- The heading has “(informative)” added, i.e. it becomes “Summary (informative)”
- The following item is amended as shown underlined;
 - By the terminal, under certain error conditions as defined in clause 6.2.2.4.

4.3.12 Ambiguities in time-shift and immediate recording

The following text in clause 6.2.2.4 is modified as shown:

After starting time-shift a terminal shall:

- Dispatch an onPlaySpeedChanged ~~RecordingEvent~~ event with a speed other than 1.0 to signal a ~~state change to state 11 "time shift mode has started" of the PVR state machine~~that time-shift has started.
- Update the ~~currentTimeShiftMode~~recordingState, playPosition and playSpeed properties of the video/broadcast object.

~~After stopping time-shift a terminal shall dispatch a RecordingEvent to signal a state change to state 0 "unrealized" of the PVR state machine.~~

4.4 Clause 7 - Formats and Protocols

4.4.1 HTTP User Agent

Add a note after the first paragraph of 7.3.2.4:

NOTE: This does not apply to HTTP requests made by the MPEG DASH player or the DRM agent.

4.4.2 Editorial correction

Change the first sentence of 7.3.2.5 to:

HTTP redirects as defined in [HTTP] in response to a an HTTP request shall be supported as described in this clause.

4.4.3 Remove ApplicationSpecificDescriptor

Clause 7.2.3 is changed as follows;

- In Table 7 “Contents of XML AIT for Broadcast-independent applications”, the row for “applicationSpecificDescriptor” is changed to read as follows;

applicationSpecificDescriptor/	Optional	Outside the scope of the present document.
--------------------------------	----------	--

- The following text at the end of the clause is removed.

TS 102 809 [3] requires the definition of an "application specific descriptor" which is not used by the present document.

The following applicationSpecificDescriptor shall be supported;

```
<xs:complexType name="HBBTVApplicationSpecificDescriptor">
  <xs:complexContent>
    <xs:extension base="mis:ApplicationSpecificDescriptor">
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

4.4.4 Clarification on DASH video resolutions

The following shall be added at the end of clause 7.3.1.3.

Different requirements on video resolutions apply to content delivered using MPEG DASH as defined in Annex E.

4.4.5 Application types in the AIT

In Table 5: “Supported application signalling features”, in the row "5.3.4 Application Information Table", the following text shall be added to the “Notes” column;

Terminals shall ignore AIT sub-tables within the selected service which have an application_type that the terminal cannot decode.

4.4.6 Multiple sections for AIT HbbTV subtable

The following paragraph is added to clause 7.2.3.1 immediately before Table 5: “Supported application signalling features”.

Terminals shall support AIT subtables for HbbTV applications, i.e. that have an application type 0x10, with at least 8 sections.

4.4.7 Support for previous versions of the specification

The following is added to the cell in the “Notes” column of Table 5: “Supported application signalling features” in the row corresponding to clause 5.2.5 of TS 102 809.

Additionally terminals shall launch applications signalled with the following values for major, minor and micro - 1,1,1 and run them as defined by the requirements in the present document.

4.4.8 Ignoring unexpected tables

The following is added to clause 7.2.3.1 immediately before Table 5: “Supported application signalling features”.

Elementary streams that are used to carry an application information table may additionally carry information using other table_ids. When acquiring and monitoring for AIT elementary streams, terminals shall silently ignore table_ids not supported for carriage of AIT information.

NOTE: The present document only requires support for table_id 0x74 as defined in TS 102 809.

The following is added to the end of clause 7.2.2.

The elementary streams used to carry DSM-CC object carousel sections may additionally carry information using other table_ids. When acquiring and monitoring for DSM-CC object carousel sections, terminals shall silently ignore table_ids not supported for carriage of DSM-CC object carousel information.

NOTE: The present document only requires support for table_id 0x3b, 0x3c or 0x3d as defined in ISO/IEC 13818-6.

4.4.9 XML Parsing

The following is added to clause 7.2.3.2 between the first and second paragraphs (before "The semantics of the fields ..."):

```
The XML AIT must not contain an XML Document Type Definition
("&lt;!DOCTYPE ...&gt;").
```

The following is added to clause 7.2.4 at the end of the clause;

```
The XML event description file defined in clause 8.2 of TS
102 809[3] must not contain an XML Document Type Definition
("&lt;!DOCTYPE ...&gt;").
```

4.4.10 Transcoding of audio metadata

The following is added at the end of clause 7.3.1.4 following the paragraph beginning "The terminal shall use metadata, where provided, to control the stereo down-mix from multichannel audio".

The remaining text in this clause applies to normal transcoding and does not consider cases where application or system sounds are inserted in addition.

If AC-3 is used to output audio over S/PDIF, HDMI or similar transfer protocols, terminals shall transcode the available metadata of an incoming HE-AAC or E-AC-3 audio stream to match the constraints of the AC-3 bit stream syntax.

This proceeding applies to normal transcoding and does not consider cases where application or system sounds are inserted in addition. Incoming metadata parameters with values exceeding the range or granularity of the corresponding parameters in AC-3 shall be rounded to the closest value creating a lower audio output level where possible to meet the range and granularity limitations of the AC-3 bit stream syntax.

Table 9a and 9b list the ranges and granularity of the relevant parameters. The metadata transformed in order to meet the limitations of the subsequent AC-3 audio format may also be applied on the local PCM outputs of a receiver. Potential side-effects of such proceeding e.g. an impact on artistic intent should be carefully considered.

Examples for mapping of parameters:

1. HE-AAC prog_ref_level of -21.75dB mapped to AC-3 dialnorm of -21dB
2. HE-AAC prog_ref_level of -31.75dB mapped to AC-3 dialnorm of -31dB
3. E-AC-3 lorocmixlev of $-\infty$ dB mapped to AC-3 cmixlev of -6dB
4. E-AC-3 lorosurmixlev of -4.5dB mapped to AC-3 surmixlev of -6dB

If the AC-3 encoder supports Annex D of ETSI TS 102 366, E-AC-3 downmix coefficients are fully supported. HE-AAC downmix coefficients may be mapped to lorocmixlev and lorosurmixlev.

The AC-3 metadata parameters ltrcmixlev and ltrtsurmixlev as defined in Annex D of ETSI TS 102 366 have no corresponding parameters in HE-AAC. If the AC-3 encoder supports Annex D of ETSI TS 102 366 the default value for ltrtsurmixlev and ltrcmixlev is -3 dB.

Legacy AC-3 decoders that do not support Annex D of ETSI TS 102 366 ignore

lorocmixlev/lorosurmixlev and ltrtcmixlev/ltrtsurmixlev and use cmixlev/surmixlev instead.

4.4.11 Object carousel caching behaviour

In clause 7.2.2, firstly the third paragraph is extended as shown underlined.

Support for the `caching_priority_descriptor` as defined in clause B.2.2.4.2 of ETSI TS 102 809 [3] is not included. Clause B.5.2 of ETSI TS 102 809 [3] specifies that transparent caching is the default caching level in the absence of this descriptor.

Secondly the following text is added at the end of the clause.

The terminal shall consider cached information to remain valid only whilst the relevant object carousel is mounted and is being monitored. This prevents the possibility of retrieving stale data from a carousel which has been unmounted and remounted if the version number of an object has been incremented such that it has the same value as when it was cached. For the avoidance of doubt, changes to DSI messages shall not be considered to be an unmounting of the carousel.

The terminal shall consider cached information to remain valid only whilst the relevant PMT that signals the carousel is being monitored. The cache ceases to be valid if the carousel signalling is removed from the PMT.

The validity of any cached information is dependent only on the relevant object carousel and is independent of the lifecycle of any application, including applications delivered within that carousel.

Any cached information that is invalid shall be flushed from the cache.

The cache ceases to be valid when the selected broadcast service changes unless the new service contains the same carousel as the previous service (see clause B.2.10 of ETSI TS 102 809 [3]) and the terminal is able to monitor the carousel continuously.

4.4.12 Constructing the Base URL from the AIT

In clause 7.2.3.1, in table 5, in the row for “5.3.6 Transport protocol descriptors”, the following text is added to the Notes column.

When the `protocol_id` is 0x0003, only the simplified form (as defined in TS 102 809) shall be supported.

4.4.13 Signed / trusted apps

In clause 7.2.3.1, in table 5, in the row for “5.2.3 Application identification”, the entire contents of the Notes column are replaced with the following;

The value of the `application_id` has no significance for whether an application is trusted or not - see clause 11.1 for more information.

4.4.14 HTTPS for MPEG DASH

In clause 7.3.1.2, the following paragraph is extended as shown underlined.

Bitrates of up to 8 MBit/sec for the stream (including protocol overheads, e.g. TCP and HTTP) shall be supported both for delivery using regular HTTP and, in the case of DASH only, using HTTP on TLS.

4.5 Clause 8 – Browser Application Environment

4.5.1 Mismatch in removeStreamEventListener arguments

Change the definition of the removeStreamEventListener method in section 8.2.1.1 to:

```
void removeStreamEventListener(String targetURL eventURL, String eventName, EventListener listener)
```

4.5.2 StreamEvent event description

Change the description of the status property in section 8.2.1.2 to:

Equal to "trigger" when the event is dispatched in response to a trigger in the stream or "error" when an error occurred (e.g. attempting to add a listener for an event that does not exist, or when a StreamEvent object with registered listeners is removed from the carousel).

Circumstances under which an event shall be dispatched with an error status include ~~An event might be dispatched with an error status if:~~

- the StreamEvent object pointed to by targetURL is not found in the carousel or via broadband;
- the StreamEvent object pointed to by targetURL does not contain the event specified by the eventName parameter;
- the carousel cannot be mounted;
- the elementary stream which contains the StreamEvent event descriptor is no longer being monitored (e.g. due to another monitoring request or because it disappears from the PMT).

Once an error is dispatched, the listener is automatically unregistered by the terminal.

4.5.3 Mandation of stream event listeners

In clause 8.2.1.1, in the description of the eventName argument to the addStreamEventListener and removeStreamEventListener methods, "should" is changed to "shall".

4.5.4 statusText attribute of XHR with DSMCC

In clause 8.2.2, in the line referring to statusText, "Set to an empty string" shall be replaced with "Implementation dependent".

4.5.5 XML parsing

In clause 8.2.1.1, the description of the targetURL argument of the addStreamEventListener method shall be amended with additional text as shown underlined below.

The URL of the DSM-CC StreamEvent object or an HTTP or HTTPS URL referring to an XML event description file (as defined in clause 8.2 of [3] and profiled in section 7.2.4) describing the event.

4.6 Clause 9 – System Integration

4.6.1 Clarification of URL example

Change the text of the third sub-bullet of examples of URLs:

The application is signaled in a DSMCC Carousel with a Component Tag of 4 and a Base URL of /index.phphtml?param1=value1 and the current service location is dvb://1.2.3

createApplication URL: dvb://current.ait/1.1?param2=value2#foo

Resulting URL: dvb://1.2.3.4/index.phphtml?param1=value1¶m2=value2#foo

Add new text at the end of the section:

NOTE: Some browsers may use the filename suffix as a means for detecting the content type for files (other than documents - see A.2.6.2) not served via HTTP. Application authors should be careful about filename suffixes used as incorrect suffixes may result in unexpected behaviour.

4.6.2 Trick modes with adaptive bitrate content

Change the text in section 9.4:

~~If the terminal supports trick modes, the behaviour defined in clause A.2.3 shall be supported including the generation of a PlaySpeedChanged event reporting the actual speed of fast forwards and fast rewind. It is optional for a terminal to support trick modes for adaptive bitrate content.~~

4.6.3 DASH technical corrigendum

In clause 9.4, the references to “DASH [29]” shall all be replaced with references to “DASH [29] and the DASH Corrigendum [37]”.

4.6.4 Ambiguity when stopping broadband video

The following text shall be added after the third bullet in clause 9.1.1.1.

NOTE: An AV Control object that is not presenting video can obscure other parts of the application UI, including video being presented by other elements in the application or in the background. “Set to an empty string” shall be replaced with “Implementation dependent”.

4.6.5 HTTPS support for non-adaptive streaming is not required

The following text is added at the end of clause 9.1.1.2;

There is no requirement for terminals to support HTTP on TLS for non-adaptive HTTP streaming.

NOTE: The present document inherits a requirement to support HTTP on TLS for adaptive streaming from the MPEG-DASH specification [29]. However, application authors should note that DASH over TLS may not be supported on early implementations of the present document.

At the end of the first paragraph of clause 9.2, “except that support for https: is not required for non-adaptive HTTP streaming” is added as shown.

The http: and https: URL schemes shall be supported as defined in clause 8.3 of the OIPF DAE specification [1] except that support for https: is not required for non-adaptive HTTP streaming.

4.6.6 DASH audio adaptation sets and language attribute

Clause 9.4 is modified as shown.

- The language property shall be set to the contents of the lang attribute in the MPD - even when this is not a valid ISO 639.2 language code ~~from value of the lang attribute in the MPD - whether set explicitly for that component or inherited. If the lang attribute in the MPD is not set for a media content component then the value of the language property in the corresponding AVComponent class shall be Undefined.~~ The contents of the language field in the media header "mdhd" of the track shall be ignored.

NOTE: MPEG DASH requires the lang attribute to comply with RFC5646 which gives preference to ISO 639-1 2-character language codes above ISO 639-2 3-character language codes where both are defined for a language.

4.6.7 Definition of "trickmodes" for DASH content

In clause 9.4, the following text;

It is optional for a terminal to support trick modes for adaptive bitrate content.

is modified as shown;

It is optional for a terminal to support play speeds other than 0 or 1 ~~trick modes~~ for adaptive bitrate content.

4.6.8 CORS and the dvb: protocol

The following is added at the end of clause 9.2.

- Access to the content of a file delivered in a carousel shall not be blocked for violating the CORS security policy

4.6.9 AVComponent.encoding property

In clause 9.4, the requirements when media content components are delivered using DASH are extended as shown underlined;

When media content components are delivered using DASH:

- Instances of the AVComponent class shall refer to AdaptationSets carrying audio, video or subtitles.
- The componentTag shall be the value of the id attribute on the Adaptation Set (if provided).
- The encoding shall be the value of the @codec attribute on the Adaptation Set (if provided).

When an instance of the AVComponent class refers to a DASH audio media content component:

4.6.10 Seek accuracy

A new clause 9.1.3 is added as shown.

9.1.3 Seek accuracy

The play position of media content being presented by a video/broadcast object or an AV control object can be controlled using the appropriate seek API.

The information available to the terminal to assist with navigating to a requested seek point depends on the container format and protocol being used. In order to ensure that terminals can always perform a seek with reasonable speed, the following accuracy requirements are defined.

Protocol and system format	Accuracy requirements	Comments
MPEG DASH / ISO BMFF	Seeks to a position shall be performed precisely if: a) the position is within a live Period and is identifiable from the MPD as being the start of a media segment, or	Seeking accurately to a position that is not the start of a segment or subsegment will typically require the terminal firstly to identify the preceding 'sync sample' in the media segment

	<p>b) the position is within an on-demand Period and is identifiable from the Segment Index as being the start of a subsegment</p> <p>Seeks to other positions <i>should</i> position the media object precisely at the requested position. If they do not, they <i>shall</i> position the media object at the nearest position for which precise seeking is required that preserves the seek direction.</p> <p>For the definitions of live and on-demand Periods see ETSI TS 103 285 [45] clause 4.2.</p>	<p>(see clause 8.6.2 of 14496-12:2012 [31]) and then to decode but not display the frames from there leading up to the requested position.</p>
HTTP streaming / ISO BMFF	<p>Seeks to a position that is identified as a ‘sync sample’ (see clause 8.6.2 of 14496-12:2012 [31]) shall be performed precisely.</p> <p>Seeks to other positions <i>should</i> position the media object precisely at the requested position. If they do not, they <i>shall</i> position the media object at the nearest identifiable sync sample that preserves the seek direction.</p>	<p>Seeking accurately to a position that is not a ‘sync sample’ will typically require the terminal firstly to identify the preceding ‘sync sample’ in the media and then to decode but not display the frames from there leading up to the requested position.</p>
HTTP streaming / MPEG-2 TS	<p>Seeks <i>shall</i> preserve the seek direction and <i>shall</i> result in a position error no greater than one GOP length plus 20% of the interval between the last position and the requested position, provided that the media meets the following requirements:</p> <ul style="list-style-type: none"> • the bitrate averaged over any individual GOP is within +/- 10% of the bitrate averaged over the entire media asset • there are no PCR discontinuities anywhere in the media asset 	<p>This requirement is intended to permit the terminal to determine byte positions for seek points based on a reasonable measurement of the average stream bitrate, and then to commence playback at the next I-frame.</p>

Broadcast timeshift / MPEG-2 TS	Seeks <i>shall</i> preserve the seek direction and shall position the media object to within 5 seconds or one GOP length of the requested position, whichever is the greater.	The highly variable bitrate of typical broadcast services is likely to require indexing of the timeshift buffer to achieve this.
---------------------------------	---	--

In all cases, the position on the media timeline reported by the appropriate APIs shall meet the requirements specified for those APIs and shall reflect the true media position. This may mean that the position reported following a seek is different to the position requested in the seek call.

4.7 Clause 10 – Capabilities

4.7.1 Video scaling and visibility of decoded video

Change Table 11 in section 10.2.1 to:

Video scaling	<p>Terminals shall be able to present video at sizes down to 1/8 by 1/8 of the width and height of the logical video plane - equivalent to 160 x 90 pixels in the Hybrid Broadcast Broadband TV application graphics plane.</p> <p>Terminals shall be able to scale video down to 1/4 by 1/4 and should be able to scale video down to 1/8 by 1/8. For sizes between 1/4 by 1/4 and 1/8 by 1/8, terminals which cannot scale video shall crop the video instead and display it centered in the according video object of the Hybrid Broadcast Broadband TV application graphics plane.</p> <p>Terminals shall be able to scale video up to 2 x 2 of the width and height of the logical video plane.</p> <p>Within these limits, any arbitrary scaling factor shall be allowed. The aspect ratio of decoded video shall be preserved such that all of the decoded video is visible within the area of the video/broadcast or AV Control object. The aspect ratio of the video shall be preserved at all scaling factors.</p> <p>See OIPF DAE Annex H.2 for more information.</p>	
---------------	---	--

4.7.2 Errors in the example XML capabilities

Change the xml text in section 10.2.4 to:

```
<profilelist xmlns="urn:oipf:config:oitf:oitfCapabilities:2009-1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oipf:config:oitf:oitfCapabilities:2011-1 config-oitf-
oitfCapabilities.xsd">
  <ui_profile name="OITF_HD_UIPROF+DVB_S+TRICK_MODE+TRICKMODE">
    <ext>
      <parentalcontrol schemes="dvd-si">true</parentalcontrol>
      <clientMetadata type="dvd-si">true</clientMetadata>
    </ext>
  </ui_profile>
  <clientMetadata type="dvd-si">true</clientMetadata>
  <video_profile name="MP4_AVC_SD_25_HEAAC" type="video/mp4" transport="dash"/>
  <video_profile name="MP4_AVC_HD_25_HEAAC" type="video/mp4" transport="dash"/>
```

```

<audio_profile name="MPEG1_L3" type="audio/mpeg"/>
<audio_profile name="HEAAC" type="audio/mp4"/>
<video_profile name="MP4_AVC_SD_25_HEAAC" type="video/mp4" transport="dash"/>
<video_profile name="MP4_AVC_HD_25_HEAAC" type="video/mp4" transport="dash"/>
<video_profile name="TS_AVC_SD_25_HEAAC" type="video/mpeg"/>
<video_profile name="TS_AVC_HD_25_HEAAC" type="video/mpeg"/>
<video_profile name="MP4_AVC_SD_25_HEAAC" type="video/mp4"/>
<video_profile name="MP4_AVC_HD_25_HEAAC" type="video/mp4"/>
<audio_profile name="MPEG1_L3" type="audio/mpeg"/>
<audio_profile name="HEAAC" type="audio/mp4"/>
</profilelist>

```

4.7.3 Location of the DRM information in the XML capabilities

Modify the paragraph following Table 13 to:

The support of the DRM feature shall be indicated by the addition of one or more <drm> elements in the OIPF extension to the <profileList> as defined in section 9.3.10 and Annex F of the OIPF DAE specification [1] ~~to the end of the profilelist element in the above XML~~. For example:

4.7.4 Sample application no longer provided

Section 10.2.5 is deleted.

4.7.5 Extensions to A/V object for parental rating errors

In clauses 10.2.6.3 and 10.2.6.4, the paragraphs beginning “*If playback which was initiated by an Hybrid Broadcast Broadband TV application is blocked*” shall be extended with “*and an onParentalRatingChange event posted*”.

4.7.6 Maximum cookie size

In clause 10.2.1, in the row of Table 11 “Minimum terminal capabilities” specifying “Cookie support”, “simultaneously” is added to the text quoted below as shown underlined.

Since section 6.1 of RFC 6265 [24] does not fix strict limits, the present document fix the following minimum capabilities that terminals SHALL simultaneously support:

4.7.7 Availability of Key Events to Applications

In clause 10.2.2, the additions shown underlined below are made.

10.2.2 User input

Implementations shall provide a mechanism for the end user to generate key events as defined in table 1.

Table 1: Key events and their status

Button (for conventional remote controls)	Key event	Status	Availability
4 colour buttons (red, green, yellow, blue)	VK_RED, VK_GREEN, VK_YELLOW, VK_BLUE	Mandatory	<u>Always available to applications</u>
4 arrow buttons (up, down, left, right)	VK_UP, VK_DOWN, VK_LEFT, VK_RIGHT	Mandatory	<u>Always available to applications</u>
ENTER or OK button	VK_ENTER	Mandatory	<u>Always available to applications</u>
BACK button	VK_BACK	Mandatory	<u>Always available to applications</u>
Number keys	VK_0 to VK_9 inclusive	Mandatory	<u>May only be available to applications once activated</u>
Play, stop, pause	VK_STOP and either VK_PLAY and VK_PAUSE or VK_PLAY PAUSE	Mandatory	<u>May only be available to applications once activated</u>
Fast forward and fast rewind	VK_FAST_FWD VK_REWIND	Mandatory	<u>May only be available to applications once activated</u>
Record	VK_RECORD	Mandatory if the PVR feature is supported, otherwise optional.	<u>May only be available to applications once activated</u>
TEXT or TXT or comparable button	Not available to applications	mandatory	
2 program selection buttons (e.g. P+ and P-)	Not available to applications	Optional	
WEBTV or comparable button	Not available to applications	Optional	
EXIT or TV or comparable button	Not available to applications	Optional	

Key events which have a key code listed in the preceding table shall be available to all applications when requested through the `keySet` object. Key events which do not have a key code listed in the preceding table shall be handled by the implementation and not delivered to applications.

The availability column indicates if the key events are always available to applications or may only be available once the application has been activated. Terminals may choose to make key events listed as “May only available to applications once activated” only available to applications once the user has activated the application. For such terminals the following shall apply;

- Applications AUTOSTARTed by the terminal shall be activated when they have received a key event.
- Other applications (e.g. broadcast-independent applications or ones signalled as PRESENT) shall be activated when launched.
- The key set of an application shall only contain keys that are available to the application at that time.
- If a key set is requested that includes keys not available to an application then that part of the request shall be discarded and only any remaining part of the request relating to available keys shall be processed.

- When an application becomes activated, the key set shall not automatically change, the application needs to call `KeySet.setValue()` in order to receive key events that were not previously available to it but now are.

Otherwise terminals shall make key events listed as “May only available to applications once activated” always available to applications identically to those key events listed as “Always available to applications”.

4.7.8 Download management and manageDownload

In clause 10.2.1, in the minimum terminal capabilities table, in the row for “Download management”, the reference to “`manageDownload`” (singular) is “`manageDownloads`” (plural).

4.7.9 XML Parsing

In clause 10.2.4, the following new paragraphs are added at the end of the clause.

The “doctype” property of the “xmlCapabilities” property of the “application/oipfCapabilities” embedded object shall be null.

NOTE: This indicates that there is no Document Type Definition for the xmlCapabilities.

4.7.10 Graphics plane resolution

In clause 10.2.1, in Table 11: “Minimum terminal capabilities”, in the row for “Hybrid Broadcast Broadband TVApplication graphic plane resolution”, the contents of the Characteristic column are amended as shown:

~~The terminal shall have at least this graphics resolution. If it is physically higher than this then the resolution shall appear to the applications to be exactly~~ The graphics plane may have a higher resolution but the co-ordinate system as seen by applications shall always be 1 280 x 720 pixels. Note: this allows for higher resolution rendering of application text and images but limits the granularity with which an application can position graphics.

4.7.11 CICAM capabilities

In clause 10.2.4, the following text is amended as shown;

The support of one or more CA systems on a CICAMCI+ shall be indicated using the <drm> element defined in annex F of the OIPF DAE specification [1] and providing the protectionGateways attribute with "ci+" string. All of the CA systems exposed by the CICAM using the ca_info APDU shall be listed in this way. For example:

4.7.12 KeySet requirements for text input

The following text is added in clause 10.2.2 before the paragraph “On up, down, left, right keydown events, terminals shall choose one of the following navigation mechanisms in the priority order listed below:”

When the focus is on either i) an input element of a type that accepts text input (e.g. `type="text"`) or ii) a textarea element then all key events that can be generated by the "Text entry method" required by table 11 "Minimum terminal capabilities" (e.g. virtual keyboard)

shall be delivered to the element with focus regardless of whether those key events are in application's current KeySet.

4.7.13 Default component selection

A new clause 10.2.8 is added as follows;

10.2.8 Component selection by the terminal

It is the responsibility of the terminal to choose for presentation to the user the most appropriate default components from those available in the media object(s), based on the user's preferences (e.g. audio language). The terminal shall present to the user the default components of those component types which are selected; this selection shall also be based on user preferences (e.g. subtitles on/off).

If the components available within a presentation change and selection of one or more media type is being done by the terminal, then the terminal may choose a component, previously not selected, for presentation to the user, for example if that component fits better with the user's preferences.

4.8 Clause 11 – Security

4.8.1 TLS profile

Clause 11.2 is renumbered to 11.2.3 after new clauses 11.2.1 and 11.2.2 and before new clauses 11.2.4 and 11.2.5 all of which are as shown below. Technical changes relative to clauses 9.1.1.1 and 9.1.1.2 of the OIPF DAE specification are marked with underline and strikethrough.

11.2 TLS and root certificates

11.2.1 TLS support

HTTP over TLS as defined in RFC2818 [7] and RFC5246 [8] shall be supported for transporting application files over broadband.

TLS 1.2 (RFC5246[8]) should be supported for HTTP over TLS, if not then TLS 1.1 (RFC4346[21]) should be supported instead and if neither of those is supported then TLS 1.0 (RFC2246[20]) shall be supported instead.

NOTE: TLS 1.2 provides a much higher security level than TLS 1.0 and 1.1 so manufacturer are recommended to support it. Note also that TLS 1.0 and 1.1 are obsoleted by the TLS 1.2 specification. Future versions of the present document will require support for TLS 1.2 and omit the possibility of only supporting TLS 1.0 or 1.1.

In order to fix a known vulnerability in SSL and TLS renegotiation, terminals shall support the Renegotiation Indication Extension as specified in RFC5746 [39] for all TLS versions.

Terminals shall not negotiate sessions using SSL 3.0 or earlier.

Terminals shall support the Server Name Indication extension defined in RFC 6066 [40].

Terminals SHALL deem a TLS connection to have failed if any of the following conditions apply:

- Certificate chain fails validation as per RFC 5280 [9] section 6.
- The host name or IP address contained in the server certificate does not match the host name

or IP address requested. When verifying the host name against the server-supplied certificate, the '*' wildcard and the subjectAltName extension of type dNSName shall be supported as defined in RFC 2818 [7].

Terminals may deem a TLS connection to have failed if:

- Any signature required for certificate chain validation uses an algorithm or key size that is not required by this specification.

NOTE: This relates only to signatures that are actually required to be verified and does not cover signatures on root certificates or signatures on any additional certificates presented by the server for compatibility with older clients.

11.2.2 Cipher Suites

Terminals SHALL support the following cipher suites for all TLS versions:

- TLS_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- ~~TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA~~

Terminals SHALL NOT support 'anonymous' cipher suites or cipher suites with NULL encryption for TLS connections.

11.2.4 Signature algorithms

The algorithm requirements for signature verification are specified in the table below.

Terminals may cease to trust any signature that uses SHA-1 as the digest algorithm after 31st December 2016.

<u>Algorithm name</u>	<u>TLS identifier</u>	<u>Status</u>
<u>sha1WithRSAEncryption</u>	<u>0x0201</u>	<u>Mandatory until made optional by SHA-1 sunset specified above.</u>
<u>sha256WithRSAEncryption</u>	<u>0x0401</u>	<u>Mandatory</u>
<u>sha384WithRSAEncryption</u>	<u>0x0501</u>	<u>Mandatory</u>
<u>sha512WithRSAEncryption</u>	<u>0x0601</u>	<u>Optional</u>

11.2.5 Key sizes

Terminals shall support RSA keys with modulus size between 2048 and 4096 bits.

4.9 Annex A – OIPF DAE Specification Profile

4.9.1 getSIDescriptors optional argument

Change Table A.1 in Annex A.1 to:

Basics	7.16.2.1, 7.16.2.2	M(*)	<p>The following properties are required:</p> <ul style="list-style-type: none"> - name - programmeID - programmeIDType - description - longDescription - startTime - duration - channelID - parentalRatings <p>All other properties and methods are not included.</p> <p>The constants defined in clause 7.16.2.1 shall be supported however support for CRIDs is outside the scope of the present document.</p> <p>The following method is required for Programme objects returned by the programmes property of the video/broadcast object:</p> <ul style="list-style-type: none"> - getSIDescriptors 	Broadcast-related
Metadata extensions to Programme	7.16.2.3	NI		
DVB-SI extensions to Programme	7.16.2.4	NI M	<p><u>The optional argument descriptorTagExtension to the method getSIDescriptors is mandatory when descriptorTag is 0x7f and ignored in all other cases.</u></p>	

4.9.2 DVB-SI parental rating

Change Table A.1 in Annex A.1 to:

The ParentalRatingScheme class	7.9.2	M M(*)	<p>A scheme supporting DVB-SI age based rating shall be supported. <u>The threshold.value and threshold.name properties shall be undefined if the user has set no minimum age in the terminal's parental control system (i.e. the user will never be requested for their PIN) and the threshold.scheme property is dvb-si.</u></p> <p><u>The description of the "dvb-si" rating scheme in the name property is changed to:</u></p> <ul style="list-style-type: none"> <u>the string value "dvb-si": this means that the scheme of a minimum recommended age encoded as per ratings 0x01 to 0x0f in the parental rating descriptor from Error! Reference source not found. is used to represent the parental rating values.</u> <p><u>NOTE: If the broadcaster defined range from 0x10 to 0xff is used then that would be a different parental rating scheme and not "dvb-si".</u></p>	None
--------------------------------	-------	--------	---	------

4.9.3 Definition of programmeID

Change Table A.1 in Annex A.1 to:

Programme, ScheduledRecording, Recording and Download	8.4.4	M(*)	<p>Only for properties that are required by the present document.</p> <p><u>The property programmeID shall contain a DVB URI including the event id and not an integer.</u></p>	
---	-------	------	---	--

4.9.4 Extensions to video/broadcast object for recording and time-shift

Change Table A.1 in Annex A.1 to:

Extensions to video/broadcast for recording and timeshift	7.13.2	M(*), M-P NI	<p><u>OIPF DAE clause 7.13.2 shall be replaced by the text in Annex A.2.14 and A.2.15. Terminals that support the PVR feature shall support all of A.2.14 and A.2.15.</u></p> <p>Terminals that support time-shift of broadcast video shall support the following events and properties even if they do not support the full PVR option:</p> <ul style="list-style-type: none"> onRecordingEvent RecordingEvent 	
---	--------	--------------	---	--

			<ul style="list-style-type: none"> • recordingState • playPosition • <u>onPlayPositionChanged</u> • playSpeed • <u>onPlaySpeedChanged</u> • <u>playbackOffset</u> • <u>maxOffset</u> • <u>timeShiftMode</u> • <u>currentTimeShiftMode</u> 	
Extensions to AV object for trickmodes	7.14.3	M(*)	Only the <u>onPlayPositionChanged</u> property and event and <u>onPlaySpeedChanged</u> properties and events are required.	None

4.9.5 Missing channel.nid property

Change Table A.1 in Annex A.1 to:

Channel class	7.13.11	M(*)	<p>The following properties shall be supported:</p> <ul style="list-style-type: none"> - channelType - ccid - dsd - <u>nid</u> - onid - tsid - sid - name <p>All other properties and methods are not included.</p>	Broadcast-related
---------------	---------	------	---	-------------------

4.9.6 Misspelling of countryId

Change Table A.1 in Annex A.1 to:

The Configuration class	7.3.2	M(*)	<p>Support for read-only access to the following properties is mandatory:</p> <ul style="list-style-type: none"> - preferredAudioLanguage - preferredSubtitleLanguage - preferredUILanguage - countryID <u>countryId</u> <p>All other properties and methods are optional.</p>	None
-------------------------	-------	------	---	------

4.9.7 Incorrect section names

Change Table A.1 in Annex A.1 to:

IMS Related <u>Communication Services APIs</u>	7.8	NI		
---	-----	----	--	--

Broadcast video Scheduled content and hybrid tuner APIs				
video/broadcast embedded object	7.13.1	M(*)	In the <code>setChannel()</code> method, the optional <code>contentAccessDescriptorURL</code> parameter may be ignored. The <code>setVolume()</code> and <code>getVolume()</code> methods are not included. The modifications in clause A.2.4 shall be supported.	See clause A.2.4

<u>Extensions to video/broadcast for access to DVB-S EIT p/f</u>	7.13.3	M		Broadcast-related
--	--------	---	--	-------------------

<u>Extensions to video/broadcast for channel scan current channel information</u>	7.13.7	M	Access to the <code>currentChannel</code> property by broadcast-independent applications shall return <code>null</code> .	Broadcast-related
<u>Extensions to video/broadcast for creating Channel lists from SD&S fragments</u>	7.13.8	NI		
<u>ChannelConfig object class</u>	7.13.9	M(*)	The <code>channelList</code> property shall be supported. Other properties, methods and events are not included.	Broadcast-related

IMS <u>Communication Services API Support</u>	9.3.9	NI		
--	-------	----	--	--

4.9.8 Incorrect references to DAE Annex B

Change Table A.1 in Annex A.1 to:

5.4 Compatibility with CEA-2027-A	B	M		
5.4 Window scripting object changes	B	M(*)	See clause A.2.8	None
5.4 Omit Window.download()	B	M		

5.7 addition <u>modifications</u> to 5.7.1.f and 5.7.1.g	B	M		None
---	---	---	--	------

4.9.9 Missing window.onload property

Change the text in section A.2.8.1 to:

The window object shall be supported as defined in annex B of the OIPF DAE specification [1] except as follows. The following properties shall be supported on the window object:

`document`, `frames`, `history`, `innerHeight`, `innerWidth`, `location`, `name`, `navigator`, `oipObjectFactory`, `onkeypress`, `onkeydown`, `onkeyup`, `onload`, `parent`, `self`, `top`, `window`, `XMLHttpRequest`, `onblur`, `onfocus`, `frameElement`

4.9.10 Explicitly linking EIT to metadata search

Add new section A.2.9:

A.2.9 Access to EIT Schedule Information

The Metadata APIs listed in table A.1 of this document shall allow access to DVB-SI EIT event schedule information for the actual transport stream and for the other transport streams (as defined in EN 300 468 [16]) that are carried on the transport stream of the currently selected broadcast service, unless access to broadcast resources is suspended according to 6.2.2.7. The terminal shall use EIT-present/following information and, if it present, EIT-schedule information. If both EIT-schedule and EIT-present/following information are present, it is implementation dependent which shall be used in cases where there are conflicts.

4.9.11 Mandatory argument to sendDRMMMessage

Change Table A.1 in Annex A.1 to:

Content Service Protection AP	7.6	M-C, M-M	Mandatory if the DRM feature is supported or if the terminal supports CI+. The DRMSystemID argument for the sendDRMMMessage method shall be specified and shall not be null.	Trusted
-------------------------------	-----	----------	---	---------

4.9.12 Queue management on type change

Make the following change in Annex A.2.5:

Calling stop(), modifying the data and/or type property or entering the error state shall cause any queued media item to be discarded.

4.9.13 Audio from memory

Change Table A.1 in Annex A.1 to:

Media control	4.4.5	M(*)	Shall be modified as defined in clause A.2.1. In addition, all references to playing audio from memory in this clause shall not apply.	
---------------	-------	------	---	--

4.9.14 Remove use of display:none

- In clause A.2.5, the following changes shall be made;
- The current text of clause A.2.5 shall be renumbered to become A.2.5.1 by the insertion of a new heading “A.2.5.1 New queue method”.

A new clause A.2.10 shall be added as follows;

A.2.10 Modification to application visibility

Clause 4.4.6 of the OIPF DAE specification [1] suggests setting the BODY element of an application to display:none when it does not ever need to be visible. Instead of this method, it is suggested that visibility:hidden would be more appropriate.

4.9.15 Extensions to A/V object for parental rating errors

Change table A.1 in annex A.1 as shown;

Extensions to A/V object for parental rating errors	7.14.5	O-M, <u>M-D</u> , <u>M-P</u>	The description of how a particular DRM technology integrates with the present document may make this mandatory	None
---	--------	---------------------------------	---	------

4.9.16 ScheduledRecording constants not included

The following text shall be added before table A.1 in annex A.1.

Where constants are defined in in the OIPF specification as input parameters and/or return values for methods or as values for properties, these constants shall be supported if any method or property is supported that uses them and if the constant is not explicitly excluded by name above. Although the constants defined in the OIPF specification are expressed in JavaScript as properties, statements in the table above that “Only the following properties shall be supported” do not apply to these constants.

4.9.17 Copy/paste error in A.2.4.2

The following changes shall be made to the second paragraph of clause A.2.4.2.

Broadcast-related applications shall have full access to the video/broadcast object. If a new broadcast service is selected then this may result in the broadcast-related application being killed as defined in clause 6.2.2.24. As defined in clause 6.2.2.2, selecting Access to MPEG programs which are not broadcast services and which do not contain an AIT will not ~~have these consequences~~ cause the running broadcast-related application to be killed.

4.9.18 Changes to Extensions for playback of selected media components

Change Table A.1 in Annex A to;

Extensions for playback of selected media components	7.16.5	M(*)	The <code>selectComponent</code> and <code>unselectComponent</code> methods shall be asynchronous. The <code>getComponents</code> method shall always return fresh information. For example, in the case of an MPEG-2 transport stream, after a change to the PMT.	
--	--------	------	--	--

4.9.19 Origins for carousel delivered applications

The following sentence shall be added at the end of clause A.2.8.1.

The support for XMLHttpRequest shall include cross-origin resource sharing (CORS) as defined in CORS [38].

4.9.20 Changes to OIPF section 8.4.2

Change Table A.1 in Annex A to;

AVComponent	8.4.2	M(*)	Only for properties that are required by the present document. <u>Statements that a property “may” be derived in a particular way shall be read as “shall” be derived in that way.</u> For AVComponents corresponding to an MPEG DASH Adaptation Set, the Language property shall be what is encoded in the MPD which may be an ISO 639-1 2-character language code and not an ISO 639-2 3-character language code.	
-------------	-------	------	---	--

4.9.21 onPlaySpeedChanged - Should always dispatch when play() is called

Change Table A.1 in Annex A to;

State diagram for A/V control objects	7.14.1.1	M(*)	An onPlaySpeedChanged event shall be generated for all calls to the play() method regardless of the value returned by the method call and whether the play speed changes or not.	None
---------------------------------------	----------	------	--	------

4.9.22 CEA-2014 Tagged Opcodes Replacement

A new clause A.2.6.5 is be added as follows;

A.2.6.5 CEA-2014 tagged opcodes replacement

The tagged opcode mechanism defined in clause 5.4.a.9 of CEA-2014 [i.1] is not required to be supported.

4.9.23 Usage of setAttribute("data",value) method of AV HTMLObject

A new clause A.2.6.6 is added as follows;

A.2.6.6 Attribute reflection for visual embedded objects

The IDL attributes of an object element representing an AV control or video/broadcast object shall reflect the element’s content attributes of the same names respectively, as defined in sections 2.7.1, 4.8.4 and 4.8.17 of the HTML5 specification as referenced by the OIPF DAE specification [1].

Note: This means that the attributes ‘data’, ‘type’, 'name', ‘width’, and ‘height’ can be set and read either by accessing the object element’s JavaScript properties of the same names, or by invoking the object element’s setAttribute/getAttribute properties.

4.9.24 CSS display:none and OIPF objects

The following text is added to table A.1 in the row for “Object Factory API” in the “Notes” column.

Note: Application authors should use the oipfObjectFactory to access the non-visual embedded objects such as the application manager and configuration objects. When the <object> tag is used, authors should be aware that the use of the CSS property display with a

value of `none` may cause unexpected behaviour in relation to that object

4.9.25 MIME type for HTML documents

The following note is added to A.2.6.2 immediately following the paragraph whose first sentence is “All XHTML documents of an Hybrid Broadcast Broadband TV application shall be served with the MIME content type "application/vnd.hbbtv.xhtml+xml".”

NOTE: XHTML documents may be served with parameters appended to the MIME type given above.

4.9.26 TLS Profile

In clause A.1, the row of the table corresponding to clause 9.1.1 of the OIPF DAE specification is amended as shown;

SSL/TLS Requirements	9.1.1	M (*)NI	9.1.1.1 and 9.1.1.2 are required. 9.1.1.3 is r Replaced by clause 11.2 of the present document.	
----------------------	-------	--------------------	--	--

4.9.27 Encoding parental rating for dvb-si

In clause A.1, the row of the table corresponding to clause 7.9.4 of the OIPF DAE specification is amended as shown.

The ParentalRating class	7.9.4	M	<u>For instances with a scheme of "dvb-si", the name property is a string containing an age in years, encoded as a decimal in the range "4" to "18" inclusive. For example, "13" means a programme that is rated suitable for persons of 13 years of age or older.</u>	None
--------------------------	-------	---	--	------

In clause A.1, the row of the table corresponding to clause E.3 of the OIPF DAE specification is amended as shown.

Abstract Content Access Descriptor Format	E.3	M-D, O-M	<p>Shall be supported if the download features is supported. The description of how a particular DRM technology integrates with the present document may make this mandatory.</p> <p><u>When parsing a <ParentalRating> element, the content of that element is used as the name property of the JavaScript ParentalRating object. Valid values are the ones defined as valid for a ParentalRating object using the indicated scheme.</u></p>
---	-----	----------	---

4.9.28 JavaScript classes being addressed as "window."

The following methods shall be supported on the window object:

close(), debug(),setTimeout(), setInterval(), clearTimeout(), clearInterval(),addEventListener(), removeEventListener()

NOTE: The JavaScript language specification requires that a JavaScript classes can be addressed with a "window." prefix identically to the behaviour as if the prefix was not present. e.g. since KeyEvent.VK_ENTER is required, window.KeyEvent.VK_ENTER is also required and the two are identical.

All other methods and properties are not included.

4.9.29 XML Parsing

1) In clause A.1 "Detailed section by section definition":

In the row "Content Access Download Descriptor Format", add to the "Notes" column:

The Content Access Download Descriptor must not contain an XML Document Type Definition ("`<!DOCTYPE ...>`").

In the row "Content Access Streaming Descriptor Format", add to the "Notes" column after the existing text:

The Content Access Streaming Descriptor must not contain an XML Document Type Definition ("`<!DOCTYPE ...>`").

2) In clause A.2.6.2 "MIME type and DOCTYPE", after the bulleted list, add a new paragraph:

The "doctype" declaration must not contain an "intSubset" as that is defined in the XML specification [41].

4.9.30 AVAudioComponent.audioChannels

A new clause A.2.11 is added as follows;

A.2.11 AVAudioComponent.audioChannels

In clause 7.16.5.4.1, the definition of the audioChannels property shall be extended as shown underlined.

Indicates the number of **main** channels present in this stream (e.g. 2 for stereo, 5 for 5.1, 7 for 7.1). Potentially available low frequency effects channels are not included in this indication.

In table A.1, in the row for clause 7.16.5 (“Extensions for playback of selected media components”) of the OIPF DAE specification, the following text is added;

The audioChannels property of the AVAudioComponent class is modified as defined in clause A.2.11.

4.9.31 AVComponent for stream type 0x04

A new clause A.2.12 is added as follows;

A.2.12 Modifications to Clause 8.4.2

In clause 8.4.2, in the row of the table for the type property, in the columns for MPEG-2 transport streams, the following item shall be extended as shown underlined;

A value of 0x03 or 0x04 or 0x11 in the stream_type field in the PMT -> AUDIO .

In table A.1, in the row for clause 8.4.2 (“AVComponent”) of the OIPF DAE specification, the following text is added;

See clause A.2.12.

4.9.32 programmeID in the ScheduledRecording class

In clause A.1 "Detailed section by section definition", in the row "The ScheduledRecording class", programmeID shall be added to the list of properties that shall be supported.

4.9.33 recordings property in oipfRecordingScheduler

In table A.1, in the row for "Extension to application/oipfRecordingScheduler for control of recordings", the text in the notes cell is changed as shown underlined.

The recordings property shall be supported and shall return recordings that are in-progress as well as ones that are scheduled or completed. Other properties, methods and events are not included.

4.9.34 Modifications to Clause 7.10.1.1 and references to it

A new clause A.2.13 is added as follows;

A.2.13 Modifications to Clause 7.10.1.1 and references to it

In clause 7.10.1.1;

Firstly the description of the getScheduledRecordings method is modified as shown;

ScheduledRecordingCollection getScheduledRecordings()	
Description	Returns a subset of all the <u>those</u> recordings that are scheduled but which have not yet started and which. The subset SHALL include only scheduled recordings that were scheduled using a service from the same FQDN as the domain of the service that calls the method by an application from the same origin as the caller.

Secondly the description of the remove method is modified as follows;

void remove(ScheduledRecording recording)			
Description	<p>Remove a recording (either scheduled, in-progress or completed). For non-privileged applications, recordings SHALL only be removed when they are scheduled but not yet started and the recording was scheduled by the current service. Recordings SHALL only be removed when they are scheduled but not yet started. <u>Additionally for terminals with the attribute manageRecordings set in the <recording> element of their capabilities set to "samedomain", recordings shall only be removed when the recording was scheduled by applications from the same origin as the caller.</u> As with the record method, only the programmeID property of the scheduled recording SHALL be used to identify the scheduled recording to remove where this property is available. The other data contained in the scheduled recording SHALL NOT be used when removing a recording scheduled using methods other than recordAt(). For recordings scheduled using recordAt(), the data used to identify the recording to remove is implementation dependent. If an A/V control object is presenting the indicated recording then the state of the A/V Control object SHALL be automatically changed to 6 (the error state).</p>		
Arguments	<table border="1"> <tr> <td>recording</td> <td>The scheduled recording to be removed.</td> </tr> </table>	recording	The scheduled recording to be removed.
recording	The scheduled recording to be removed.		

Thirdly the `getInProgressRecordings()` method is added as shown;

ScheduledRecordingCollection getInProgressRecordings()	
Description	Returns those recordings that are currently in progress (i.e. those where recording has started but has not yet completed) and which were scheduled by an application from the same origin as the caller.

In clause 9.3.3, the following two modifications are made as shown;

The Boolean attribute `manageRecordings` specifies whether or not the OITF supports managing recordings through the JavaScript APIs defined in section 7.10.4 and 7.10.1.

“samedomain”: indicates that recordings initiated by applications from the same ~~fully-qualified domain~~ origin may be managed

4.9.35 DRM API restriction to 'Trusted' applications

The following rows in table A.1 have “Trusted” replaced by “None” as shown.

The DRMControllInformation class	7.4.6	M-D+ M-M	Mandatory if both Download and DRM features are supported - even if the supported DRM systems do not use the <DRMControllInformation> element inside the content access download descriptor.	Trusted None
The DRMControllInfoCollection class	7.4.7	M-D+ M-M	If the Download feature is supported and the terminal supports CI+ and if the terminal is capable of providing downloaded content to the CI+ CAM then these classes shall be supported - even if the CAS brought by a CI+ CAM do not use the <DRMControllInformation> element inside the content access download descriptor.	Trusted None
Content On Demand Metadata APIs	7.5	NI		
Content Service Protection API	7.6	M-C, M-M	Mandatory if the DRM feature is supported or if the terminal supports CI+.	Trusted None

4.9.36 CI and video blanking by terminal

The following changes are made in clause A;

- Annex A is renamed from “OIPF DAE Specification Profile” to “OIPF Specification Profile”.
- Clause A.1 is renamed from “Detailed section by section definition” to “Detailed section by section definition for volume 5”
- Clause A.2 is renamed from “Modifications, extensions and clarifications” to “Modifications, extensions and clarifications for volume 5”.
- A new clause A.3 with sub-clause A.3.1 is added as follows;

A.3.1 Host blanking requirement

Clause 4.2.3.4.1.1.5 of the OIPF CSP specification is revised as shown;

If the program is no longer being descrambled (oipf_access_status=0), the ~~OITF SHALL blank the video decoder output.~~ The native or DAE application SHOULD not stop playing the program, as the program may become descrambled again later (access criteria change, parental unlocking etc).

~~If the program being played is descrambled again (oipf_access_status=1), the OITF SHALL display the video again.~~

Clause 4.2.3.4.1.1.6 of the OIPF CSP specification is similarly revised as shown.

If the program is no longer being descrambled (oipf_access_status=0), the ~~OITF SHALL blank the video decoder output.~~ The native or DAE application SHOULD not stop playing the program, as the program may become descrambled again later (access criteria change, rights update etc).

~~If the program being played is descrambled again (oipf_access_status=1), the OITF SHALL display the video again.~~

4.9.37 CI parental_control_info message

A new clause A.3.2 is added as follows;

A.3.2 Processing of the CI parental_control_info message

Section 4.2.3.4.1.1.5 shall be modified as shown underlined below:

When the parental_control_info message is received and a DAE application is launched, the OITF SHALL issue the relevant event to the DAE application:

- onParentalRatingChange event, if the parental rating system specified by the oipf_rating_type is supported by the OITF.
- onParentalRatingError event, if the parental rating system specified by the oipf_rating_type is not supported by the OITF.

NOTE: When processing a parental_control_info message, an OITF supporting (or not) for a parental rating system is only used to determine which event is issued to a DAE application (as above) and to set the attributes of the event (as below) for supported parental rating systems. Parental rating thresholds and PIN codes set in the terminal are not used in this process and the terminal does not generate an UI.

The prototype of the onParentalRatingChange and onParentalRatingError events defined in [DAE] are recalled here:

```
function onParentalRatingChange( String contentID, ParentalRating rating, String
DRMSSystemID, Boolean blocked )
function onParentalRatingError( String contentID, ParentalRating rating, String DRMSSystemID)
```

4.9.38 On-going Recordings in the oipfRecordingScheduler object

In table A.1, in the row for "Recording extensions to Programme", "NI" is changed to "M-P".

4.9.39 onPlayStateChange and playState

The following is added at the end of new clause A.2.5.2;

All occurrences of the function 'onPlayStateChange' in clause 7.14 shall be replaced with the following:

```
function onPlayStateChange ( Number state )
```

The function that is called when the play state of the A/V control object object changes for any reason.

The specified function, when called, should include the argument state. This argument is defined as follows:

- Number state – the new state of the A/V control object. Valid values are given in the definition of the playState property [Req. 5.7.1.f].

4.9.40 No defined mimetype for A/V control object when playing back PVR

In table A.1, in the row for "Extensions to A/V Control object for playing media objects", the following text is added to the "Notes" column;

Calls to the setSource() method where id is a recording identifier shall result in the type

attribute being set to “video/mpeg” regardless of the format in which the content is recorded.

4.9.41 Defining "delivery system descriptor" for DVB S2/C2/T2

A new clause A.2.4.4 is added as shown;

A.2.4.4 Definition of “delivery system descriptor”

The definitions of the createChannelObject(Integer idType, String dsd, Integer sid) method on the video/broadcast object, and the dsd attribute on the returned Channel object, both refer to the “delivery system descriptor”. This “delivery system descriptor” shall be as follows:

For a DVB-T channel, the “delivery system descriptor” shall be a terrestrial_delivery_system_descriptor.

For a DVB-T2 channel, the “delivery system descriptor” shall be a T2_delivery_system_descriptor which shall include at least one centre_frequency field.

For a DVB-S channel, the “delivery system descriptor” shall be a satellite_delivery_system_descriptor.

For a DVB-S2 channel that is in NBC-BS mode (as that term is used in [16]), the “delivery system descriptor” shall be a satellite_delivery_system_descriptor.

For a DVB-S2 channel that is not in NBC-BS mode, the “delivery system descriptor” shall be the concatenation of a S2_satellite_delivery_system_descriptor and a satellite_delivery_system_descriptor, in that order.

For a DVB-C channel, the “delivery system descriptor” shall be a cable_delivery_system_descriptor.

For a DVB-C2 channel that does not use channel bundling, the “delivery system descriptor” shall be a C2_delivery_system_descriptor.

For a DVB-C2 channel that uses channel bundling, the “delivery system descriptor” shall be the concatenation of one or more C2_bundle_delivery_system_descriptor.

The descriptors referred to above are all defined in EN 300 468 [16].

4.9.42 Selecting media components in language not supported by terminal

In clause A.1, in the rows for “Extensions to video/broadcast for playback of selected components”(7.13.4) and “Extensions to A/V object for playback of selected components” (7.14.4) the following text is added

HbbTV terminals shall allow HbbTV applications to select media components in language(s) not supported by the terminal where there are no other reasons to refuse the selection (e.g. codec or subtitle character set not supported). For example, a terminal supporting French, German and Polish shall allow HbbTV applications to select media components in English, Italian or Chinese. This is applicable to media components delivered by broadband and media components delivered by broadcast.

4.9.43 Mixed content

A new clause A.2.6.7 is added as follows;

A.2.6.7 Mixed content

Application developers should be aware that HbbTV terminals may implement the W3C Mixed Content specification [i.7] (subject to the requirements in this clause) and should write applications such that they work correctly on such terminals.

An HbbTV terminal that implements the Mixed Content specification [i.7] shall not consider video or audio loaded via the A/V control object as blockable content for the purposes of protecting against mixed content.

4.9.44 repeatDays property from ScheduledRecording class

In clause A.1, in the row for the “application/oipfRecordingScheduler embedded object”, the contents of the “Status in Hybrid broadcast broadband TV” and “Notes” columns are changed as shown;

application/oipfRecordingScheduler embedded object	7.10.1	M-P (*)	<u>Support for repeated recordings with the recordAt method is not included and hence the repeatDays argument may be ignored.</u>	Trusted
--	--------	---------	---	---------

4.9.45 Modifications to A.2.4.1

Clause A.2.4.1 is modified as shown. In this text, additions to TS 102 796 are shown in italics. Underlining is used to indicate text that is to be underlined in TS 102 796. Strike-through text in italics is to be included in TS 102 796 also with strike-through. Strike-through text not in italics indicates text to be deleted from TS 102 796.

A.2.4.1 State machine and related changes

This clause describes a set of changes to the state machine *and following related text* for the video/broadcast object defined in clause 7.13.1-4 of the OIPF DAE specification [1].

- Calling the setChannel() method from any state of the video/broadcast object with a null argument shall cause the application to transition to a broadcast-independent application (as described in clause 6.2.2.6). This is in addition to what is required by OIPF - e.g. causing the video/broadcast object to transition to the unrealized state and releasing any resources used for decoding video and/or audio. Hence the setChannel(null) and release() methods do not have the same behaviour in the present document.
- Suspension of access to broadcast resources as defined in clause 6.2.2.7 of the present document shall be treated as a transient error.
- *A video/broadcast object with a CSS rule of display:none shall not be loaded and hence shall not be decoding audio or video.*
- *In table 12, “State transitions for the video/broadcast embedded object”, the following row is modified as shown underlined:*

<u>Old State</u>	<u>Trigger</u>	<u>New State</u>	<u>State Transition Events</u>	<u>Description</u>
Stopped	bindToCurrentChannel()	Connecting	PlayStateChange	Video and audio presentation is enabled. <u>The terminal starts to present the current channel.</u>

- In clause 7.13.1.3 of the OIPF DAE specification [1], the definition of the bindToCurrentChannel method is modified as shown:

If the video/broadcast object is in the unrealized state and ~~video from~~ exactly one channel is currently being presented by the OITF then this binds the video/broadcast object to that ~~video~~channel (even if the current channel does not contain video and/or audio). If more than one channel is currently being presented by the OITF then this binds the video/broadcast object to the channel whose audio is being presented.

If the video/broadcast object is in the stopped state then this restarts presentation of ~~video and audio from~~ the current channel under the control of the video/broadcast object. ~~If video from more than one channel is currently being presented by the OITF then this binds the video/broadcast object to the channel whose audio is being presented.~~

4.9.46 Ambiguities in time-shift and immediate recording

New clauses A.2.14 and A.2.15 are added as shown.

A.2.14 Extensions to video/broadcast for time-shift

A.2.14.1 General

If a terminal has indicated support in its capability description for recording functionality (i.e. by giving value true to the <recording> element as specified in OIPF DAE [1] clause 9.3.3), the terminal shall support the following additional constants, properties and methods on the video/broadcast object, in order to start a time-shift of the current broadcast.

Note that this functionality is subject to the security model as specified in OIPF DAE [1] clause 10.1.

Terminals may restrict access to the time-shift methods to those applications that are signalled as safe to run when time-shifting, i.e. those signalled in the AIT with an applicaton_recording_descriptor and both the trick_mode_aware_flag and the time_shift_flag set to '1' as described in clause 7.2.3.1

The properties and methods defined in this clause are used when the content presented in the video/broadcast object is being time-shifted.

A.2.14.2 Constants

Name	Value	Use
POSITION_START	0	Indicates a playback position relative to the start of the buffered content.
POSITION_CURRENT	1	Indicates a playback position relative to the current playback position.
POSITION_END	2	Indicates a playback position relative to the end of the buffered content (co-incident with the live playback position).

A.2.14.3 Properties

function **onPlaySpeedChanged**(Number speed)

The function that is called when the playback speed of a channel changes during timeshift.

The specified function is called with one argument, `speed`, which is defined as follows:

- `Number speed` – the playback speed of the media at the time the event was dispatched.

function **onPlayPositionChanged**(Integer position)

The function that is called when a change occurs in the play position of a channel due to the use of trick play functions during timeshift.

The specified function is called with one argument, `position`, which is defined as follows:

- `Integer position` – the playback position of the media at the time the event was dispatched, measured in milliseconds from the start of the timeshift buffer. If the play position cannot be determined, this argument takes the value `undefined`.

readonly Integer **playbackOffset**

Returns the playback position during timeshift, specified as the number of seconds between the live broadcast and the currently rendered position in the timeshift buffer, where a value of zero means that the broadcast is not being timeshifted or is playing from the live point in a timeshift buffer.

When the `currentTimeShiftMode` property has the value 0, the value of this property is `undefined`.

readonly Integer **maxOffset**

Returns the maximum playback offset, in seconds of the live broadcast, which is supported for the currently rendered broadcast. If the maximum offset is unknown, the value of this property shall be `undefined`.

NOTE: This value gives the size of the timeshift buffer.

When the `currentTimeShiftMode` property has the value 0, the value of this property is `undefined`.

readonly Integer **playPosition**

If the value of the `currentTimeShiftMode` property is 1, the current playback position of the media, measured in milliseconds from the start of the timeshift buffer.

readonly Number **playSpeed**

The current play speed of the media.

readonly Number playSpeeds []
<p>Returns the ordered list of playback speeds, expressed as values relative to the normal playback speed (1.0), at which the currently specified content can be played (as a time-shifted broadcast in the video/broadcast object), or undefined if the supported playback speeds are not known.</p> <p>If timeshift is supported by the terminal, the playSpeeds array shall always include at least the values 1.0 and 0.0.</p> <p>This property may include the playback speeds that this broadcast content could be played back after being recorded, but only if they also apply to playback of the content when timeshifted.</p>

Integer timeShiftMode						
<p>The time shift mode indicates the mode of operation for support of timeshift playback in the video/broadcast object. Valid values are:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Timeshift is turned off.</td> </tr> <tr> <td>1</td> <td>Timeshift shall use "local resource".</td> </tr> </tbody> </table> <p>If property is not set the default value of the property is 1.</p>	Value	Description	0	Timeshift is turned off.	1	Timeshift shall use "local resource".
Value	Description					
0	Timeshift is turned off.					
1	Timeshift shall use "local resource".					

readonly Integer currentTimeShiftMode						
<p>When timeshift is in operation the property indicates which resources are currently being used. Valid values are:</p> <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>No timeshift.</td> </tr> <tr> <td>1</td> <td>Timeshift using "local resource".</td> </tr> </tbody> </table>	Value	Description	0	No timeshift.	1	Timeshift using "local resource".
Value	Description					
0	No timeshift.					
1	Timeshift using "local resource".					

A.2.14.4 Methods

Boolean pause()		
<table border="1"> <tr> <td>Description</td> <td>Pause playback of the broadcast. This is equivalent to <code>setSpeed(0)</code>.</td> </tr> </table>	Description	Pause playback of the broadcast. This is equivalent to <code>setSpeed(0)</code> .
Description	Pause playback of the broadcast. This is equivalent to <code>setSpeed(0)</code> .	

Boolean resume()		
<table border="1"> <tr> <td>Description</td> <td>Resumes playback of the time-shifted broadcast channel. This is equivalent to <code>setSpeed(1)</code>.</td> </tr> </table>	Description	Resumes playback of the time-shifted broadcast channel. This is equivalent to <code>setSpeed(1)</code> .
Description	Resumes playback of the time-shifted broadcast channel. This is equivalent to <code>setSpeed(1)</code> .	

Boolean setSpeed (Number speed)		
Description	<p>Sets the playback speed of the time-shifted broadcast to the value <i>speed</i>. If the value of the <code>timeShiftMode</code> property is 0 or if trick play is not supported for the channel currently being rendered, this method shall return <code>false</code> and have no effect.</p> <p>If <i>speed</i> is a value less than 1.0 and the broadcast was not previously being time-shifted, this method shall start recording the broadcast that is currently being rendered live (i.e. not time-shifted) in the <code>video/broadcast</code> object. If the terminal has buffered the 'live' broadcasted content, the recording starts with the content that is currently being rendering in the <code>video/broadcast</code> object. Acquiring the necessary resources to start recording the broadcast may be an asynchronous operation, and presentation of the broadcast may not be affected until after this method returns; applications may receive updates by registering a listener for <code>PlaySpeedChanged</code> events as defined in A.2.9.5.</p> <p>If <i>speed</i> is a value greater than 1.0 and the broadcast was not previously being time-shifted, this method shall have no effect and shall return <code>false</code>.</p> <p>When playback is paused (i.e. by setting the play speed to 0), the last decoded video frame shall be shown.</p> <p>If the time-shifted broadcast cannot be played at the desired speed, specified as a value relative to the normal playback speed, the playback speed will be set to the best approximation of <i>speed</i>.</p> <p>If there is no change to the play speed as a result of the method call, it shall return <code>false</code>.</p> <p>Unless specified otherwise above, this method shall return <code>true</code>.</p> <p>After initial operation of <code>setSpeed()</code> several events may affect the content playback.</p> <p>If during fast forward the end of stream is reached the playback shall resume at normal speed and a <code>PlaySpeedChanged</code> event generated. If the end of the timeshift buffer is reached due to end of content the playback shall automatically be paused and a <code>PlaySpeedChanged</code> event generated. Any resources used for time-shifting shall not be discarded.</p> <p>If during rewinding the playback reaches the point that it cannot be rewound further, playback shall resume at normal speed and a <code>PlaySpeedChanged</code> event generated.</p> <p>A <code>PlaySpeedChanged</code> event shall be generated when the operation has completed, regardless of the success of the operation. If the operation fails, the argument of the event shall be set to the previous play speed.</p>	
Arguments	<i>speed</i>	The desired relative playback speed, specified as a float value relative to the normal playback speed of 1.0. A negative value indicates reverse playback.

Boolean seek (Integer offset, Integer reference)		
Description	<p>Sets the playback position of the time-shifted broadcast that is being rendered in the <code>video/broadcast</code> object to the position specified by the offset and the reference point as specified by one of the constants defined in A.2.9.2. Playback of live content is resumed if the new position equals the end of the time-shift buffer. Returns <code>true</code> if the playback position is a valid position to seek to, <code>false</code> otherwise. If time-shift is not supported for the current channel (e.g. due to restrictions imposed by a conditional access or DRM system) or the broadcast is not currently being time-shifted or if the position falls outside the time-shift buffer, the terminal shall ignore the request to seek and shall return the value <code>false</code>.</p> <p>Applications are not required to pause playback of the broadcast or take any other action before calling <code>seek()</code>.</p> <p>This operation may be asynchronous, and presentation of the video may not be affected until after this method returns. For this reason, a <code>PlayPositionChanged</code> event shall be generated when the operation has completed, regardless of the success of the operation. If the operation fails, the argument of the event shall be set to the previous play position.</p> <p>After initial operation of <code>seek()</code> several events may affect the content playback.</p> <p>If during this operation the live playback position is reached the playback shall resume at normal speed and a <code>PlaySpeedChanged</code> event generated. If the timeshift buffer cannot be rewound any further, the playback shall automatically be paused and a <code>PlaySpeedChanged</code> event generated. Any resources used for time-shifting shall not be discarded.</p>	
Arguments	<i>offset</i>	The offset from the reference position, in seconds. This can be either a positive value to indicate a time later than the reference position or a negative value to indicate time earlier than the reference position.
	<i>reference</i>	The reference point from which the offset shall be measured. The reference point can be either <code>POSITION_CURRENT</code> , <code>POSITION_START</code> , or <code>POSITION_END</code> .

Boolean stopTimeshift ()	
Description	<p>Stops rendering in time-shifted mode the broadcast channel in the <code>video/broadcast</code> object and, if applicable, plays the current broadcast from the live point and stops time-shifting the broadcast. The terminal may release all resources that were used to support time-shifted rendering of the broadcast</p> <p>Returns <code>true</code> if the time-shifted broadcast was successfully stopped and <code>false</code> otherwise. If the <code>video/broadcast</code> object is currently not rendering a time-shifted channel, the terminal shall ignore the request to stop the time-shift and shall return the value <code>false</code>.</p>

In addition to these methods, the terminal shall support an additional optional attribute “offset” on the `setChannel(Channel channel, Boolean trickplay, String contentAccessDescriptorURL)` method of the `video/broadcast` object as defined in OIPF DAE [1] clause 7.13.1.3, if the terminal has indicated support for scheduled content over IP by defining one or more `ID_IPTV_*` values as part of the transport attribute of the `<video_broadcast>` element in the capability description.

<pre>void setChannel(Channel channel, Boolean trickplay, String contentAccessDescriptorURL, Integer offset)</pre>		
Description	<p>Requests the terminal to switch a (logical or physical) tuner to the specified <code>channel</code> and render the received broadcast content in the area of the browser allocated for the video/broadcast object, as specified by the <code>setChannel(Channel channel, Boolean trickPlay, String contentAccessDescriptorURL)</code> method in OIPF DAE [1] clause 7.13.1.3.</p> <p>The additional <code>offset</code> attribute optionally specifies the desired offset with respect to the live broadcast in number of seconds from which the terminal should start playback immediately after the channel switch (whereby <code>offset</code> is given as a positive value for seeking to a time in the past). If an terminal cannot start playback from the desired position, as indicated by the specified <code>offset</code> (e.g. because the terminal did not, or could not, record the specified channel prior to the call to <code>setChannel</code>), if the specified <code>offset</code> is '0', or if the <code>offset</code> is not specified, the terminal shall start playback from the live position after the specified channel switch.</p>	
Arguments	<i>channel</i>	As defined for method <code>setChannel()</code> in OIPF DAE [1] clause 7.13.1.3.
	<i>trickplay</i>	Optional flag as defined for method <code>setChannel()</code> in OIPF DAE [1] clause 7.13.1.3.
	<i>contentAccessDescriptorURL</i>	Optional attribute as defined for method <code>setChannel()</code> in OIPF DAE [1] clause 7.13.1.3.
	<i>offset</i>	The optional <code>offset</code> attribute may be used to specify the desired offset with respect to the live broadcast in number of seconds from which the terminal should start playback immediately after the channel switch (whereby <code>offset</code> is given as a negative value for seeking to a time in the past).

A.2.14.5 Events

For the intrinsic events “onRecordingEvent”, “onPlaySpeedChanged” and “onPlayPositionChanged”, corresponding DOM level 2 events shall be generated, in the following manner:

Intrinsic event	Corresponding DOM 2 event	DOM 2 Event properties
onRecordingEvent	RecordingEvent	Bubbles: No Cancelable: No Context Info: state, error, recordingId

onPlaySpeedChanged	PlaySpeedChanged	Bubbles: No Cancelable: No Context Info: speed
onPlayPositionChanged	PlayPositionChanged	Bubbles: No Cancelable: No Context Info: position

Note: the DOM 2 events are directly dispatched to the event target, and will not bubble nor capture. Applications should not rely on receiving these events during the bubbling or the capturing phase. Applications that use DOM 2 event handlers shall call the `addEventListener()` method on the `video/broadcast` object itself. The third parameter of `addEventListener`, i.e. “`useCapture`”, will be ignored.

A.2.15 Extensions to video/broadcast for recording

A.2.15.1 General

If a terminal has indicated support in its capability description for recording functionality (i.e. by giving value `true` to the `<recording>` element as specified in OIPF DAE [1] clause 9.3.3), the terminal shall support the following additional constants, properties and methods on the `video/broadcast` object, in order to start a recording of the current broadcast.

Note that this functionality is subject to the security model as specified in OIPF DAE [1] clause 10.1.

The recording functionality is subject to the state transitions represented in the state diagram in Figure XX.

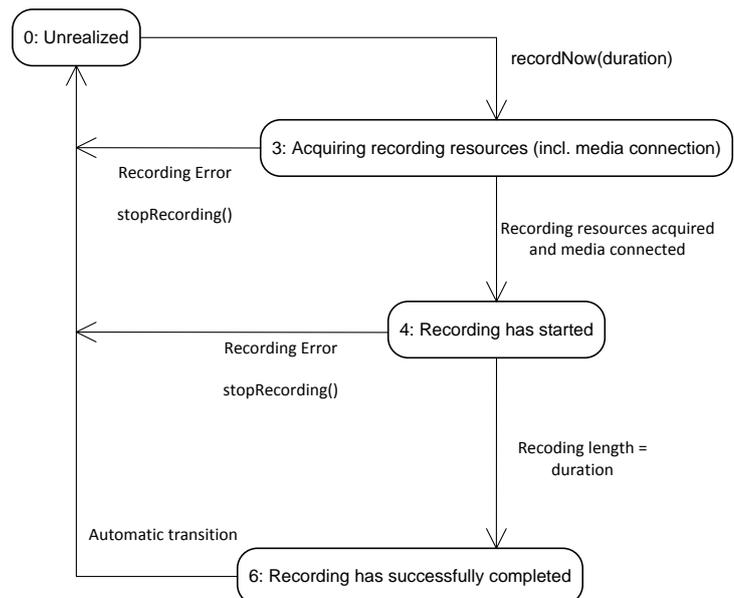


Figure 16: PVR States for `recordNow` using `video/broadcast` (normative)

Note that when the user switches to another channel whilst the current channel is being recorded using `recordNow` or the `video/broadcast` object gets destroyed, the conflict resolution and the release of resources is implementation dependent. The terminal may report a recording error using a `RecordingEvent` with value 0 (“Unrealized”) for argument state and with value 2 (“Tuner conflict”) for argument error in that case.

A.2.15.2 Properties

readonly Integer recordingState	
Returns the state of the terminal’s timeshift and <code>recordNow</code> functionality for the channel shown in the <code>video/broadcast</code> object. One of:	
Value	Description
0	Unrealized: user/application has not requested timeshift or <code>recordNow</code> functionality for the channel shown. No timeshift or recording resources are claimed in this state.
1	Value not used
2	Value not used
3	Acquiring recording resources (for example, space on the media storage device).
4	Recording has started.
5	Value not used
6	Recording has successfully completed.

function onRecordingEvent (Integer state, Integer error, String recordingId)	
This function is the DOM 0 event handler for notification of state changes of the recording functionality. The specified function is called with the following arguments:	
<ul style="list-style-type: none"> Integer state - The current state of the recording. One of: 	
Value	Description
0	Unrealized: user/application has not requested <code>recordNow</code> functionality for the channel shown. No recording resources are claimed in this state.
1	Value not used
2	Value not used
3	Acquiring recording resources (for example, space on the media storage device).

4	Recording has started.
5	Value not used
6	Recording has successfully completed.

- Integer `error` - If the state of the recording has changed due to an error, this field contains an error code detailing the type of error. One of:

Value	Description
0	The recording sub-system is unable to record due to resource limitations.
1	There is insufficient storage space available. (Some of the recording may be available).
2	Value not used
3	Recording not allowed due to DRM restrictions.
4	Recording has stopped before completion due to unknown (probably hardware) failure.

If no error has occurred, this argument shall take the value `undefined`.

- String `recordingId` - The identifier of the recording to which this event refers, This shall be equal to the value of the `id` property for the affected recording, if the event is associated with a specific recording. This shall be undefined when the value of `state` is 0.

A.2.15.3 Methods

String <code>recordNow(Integer duration)</code>	
Description	<p>Starts recording the broadcast currently rendered in the <code>video/broadcast</code> object. If the terminal has buffered the broadcasted content, the recording starts from the current playback position in the buffer, otherwise start recording the broadcast stream as soon as possible after the recording resources have been acquired. The specified duration is used by the terminal to determine the minimum duration of the recording in seconds from the current starting point.</p> <p>Calling <code>recordNow()</code> while the broadcast that is currently rendered in the <code>video/broadcast</code> object is already being recorded, shall have no effect on the recording and shall return the value <code>null</code>.</p> <p>In other cases, this method returns a <code>String</code> value representing a unique identifier to identify the recording. If the terminal provides recording management functionality through the APIs defined in OIPF DAE [1] clause 7.10.4, this shall be the value of the <code>id</code> property of the associated <code>Recording</code> object defined in OIPF DAE [1] clause 7.10.5.</p> <p>The terminal shall guarantee that recording identifiers are unique in relation to download identifiers and <code>CODAsset</code> identifiers.</p>

	<p>The method returns undefined if the given argument is not accepted to trigger a recording.</p> <p>If the terminal supports metadata processing in the terminal, the fields of the resulting Recording object may be populated using metadata retrieved by the terminal. Otherwise, the values of these fields shall be implementation-dependent</p>	
Argument s	<i>duration</i>	<p>The minimum duration of the recording in seconds. A value of -1 indicates that the recording should continue until stopRecording() is called, storage space is exhausted, or an error occurs. In this case it is essential that stopRecording() is called later.</p>

<code>void stopRecording()</code>	
Description	Stops the current recording started by recordNow().

4.9.47 Trick modes definition is vague for play() calls with non-DASH media

New clauses are added as shown.

A.2.4.5 Other modifications to the video/broadcast object

In clause 7.13.2.2, the definition of the property onPlayPositionChanged(Integer position) is changed as shown;
 The function that is called when change occurs in the play position of a channel due to the use of ~~trick play functions~~ random access.

A.2.5.3 Other modifications to the A/V control object

In clause 7.14.3.1 the definition of the property onPlayPositionChanged(Integer position) is changed as shown;
 The function that is called when change occurs in the play position of the media due to the use of ~~trick play functions~~ random access.

4.10 Annex B - Support for protected content delivered via broadband

4.10.1 Flags=0 is wrong

In clause B.2.5.1, in the definition of the syntax of the SampleEncryptionBox, “flags=0” shall read “flags”.

4.10.2 Broken reference in annex B

In clause B.1, the reference to “table 10 of the OIPF Metadata specification [18];” shall be replaced with “table 6 (“DRMControlInformation Type Semantics”) of the OIPF Metadata specification [18];”

4.11 Annex E – Profiles of MPEG DASH

4.11.1 Dynamic MPD served with cache control off

Add new text at the end of Annex E.2.1:

If a dynamic MPD is not served with "Cache-Control: no-cache" then terminals may not be able to acquire the updated version.

4.11.2 Missing text in E.4.2.1

Between the first set of numbered bullets and the lettered bullets, add the following text:

Notes:

4.11.3 Capitalisation of roles incorrect

In table E.2, “Role and Accessibility descriptor values for Audio Description”, in the “Role descriptor” column, “Alternate” shall be “alternate” and “Commentary” shall be “commentary”.

4.11.4 DASH video resolutions

In clause E.4.2.1, the following text;

4) Between Representations with the same full-screen resolution but different luminance resolutions as defined in table 9 "Table 9: Resolutions for Full-screen Display from 25 Hz H.264/AVC SDTV IRD and supported by 25 Hz H.264/AVC HDTV IRD, 50 Hz H.264/AVC HDTV IRD, 25 Hz SVC HDTV IRD and 50 Hz SVC HDTV IRD" and table 12 "Resolutions for Full-screen Display from H.264/AVC HDTV IRD and SVC HDTV IRD" of TS 101 154 [] (e.g. 1 920 × 1 080 and 1 440 × 1 080) (note b).

Shall be replaced as follows;

4) Between Representations with the same full-screen resolution but different luminance resolutions as follows
- as defined in clause 10.3 of the DVB DASH profile [36]
- The resolution of 720x576 for interlaced content only

4.11.5 Segment length

In E.3.2, “*Segments shall be at least 1s long, except for the last segment in an MPD which may be shorter*” shall be replaced with “*Segments shall be at least 1s long, except for the last segment of a Period which may be shorter.*”

4.11.6 Switching between representations

In clauses E.4.2.1 and E.4.2.2, “in the same Adaptation Set” shall be added as shown. In E.4.2.1;

During playback of adaptively streamed content encoded using AVC, terminals shall support transitions between video Representations in the same Adaptation Set as follows:

Likewise in E.4.2.2:

During playback of adaptively streamed content encoded using HE-AAC or E-AC3, terminals shall support transitions between audio Representations in the same Adaptation Set as follows:

4.11.7 URI for EAC-3 AudioChannelConfiguration

In clause E.2.5, in the paragraph concerning E-AC-3, "tag:dolby.com,2014:dash:audio_channel_configuration:2011" is inserted as shown.

For E-AC-3 the Audio Channel Configuration shall use ~~the~~either "tag:dolby.com,2014:dash:audio_channel_configuration:2011" or the legacy "urn:dolby:dash:audio_channel_configuration:2011" schemeURI.

4.11.8 XML Parsing

In section E.2.1 "Profile definition", after the last bullet point, the following bullet point is added;

The MPD must not contain an XML Document Type Definition ("<!DOCTYPE ...>")

4.11.9 Ignore MPD, Adaptation Set or Representation where no supported profile is used

Clause E.4.1 is extended as shown.

Terminals shall support the HbbTV ISOBMFF Live profile. Other profiles (e.g. the DASH-IF DASH-AVC/264 main interoperability point (see [i.8])) may be supported.

The following rules apply for MPDs that do not list "urn:hbbtv:dash:profile:isoff-live:2012" in their MPD@profiles attribute:

- MPDs specified with the "urn:mpeg:dash:profile:isoff-live:2011" profile should be supported.
- MPDs specified with the "urn:mpeg:dash:profile:isoff-on-demand:2011" profile shall be rejected by HbbTV terminals that do not support the DASH on demand profile.
- MPDs specified with the URNs defined for the interoperability points defined in the DASH-IF guidelines [i.8] shall be rejected by HbbTV terminals that do not support the specified inter-operability point.
- MPDs specified with profiles beginning "urn:hbbtv:dash:profile" shall be rejected unless that profile is defined in a later version of the present document and the HbbTV terminal supports the specified profile.
- MPDs specified with profiles beginning "urn:dvb:dash:profile" shall be rejected by HbbTV terminals that do not support the specified profile.
- If an MPD specifies multiple profiles (but not the one required to be supported by the present document) where some of them are required to be rejected by the rules in this clause and others are not required to be rejected by those same rules then the MPD is not required to be rejected.

The following rules apply for Adaptation Sets and/or Representations that are not indicated as conforming to the "urn:hbbtv:dash:profile:isoff-live:2012" profile:

- Adaptation Sets or Representations indicated as being compliant with "urn:mpeg:dash:profile:isoff-on-demand:2011" shall be ignored by HbbTV terminals that do not support the DASH on demand profile.
- Adaptation Sets or Representations indicated as being compliant with "urn:mpeg:dash:profile:isoff-live:2011" should not be ignored unless there are other reasons to do so (e.g. non-supported codec or @role).

- Adaptation Sets or Representations indicated as being compliant with one or more of the interoperability points in the DASH-IF interoperability guidelines [i.8] shall be ignored by HbbTV terminals that do not support that inter-operability point.
- Adaptation Sets or Representations indicated as being compliant with profiles beginning "urn:hbbtv:dash:profile" shall be ignored unless the indicated profile is defined in a later version of the present document and the HbbTV terminal supports that profile.
- Adaptation Sets or Representations indicated as being compliant with profiles beginning "urn:dvb:dash:profile" shall be ignored by HbbTV terminals that do not support the indicated profile.
- Adaptation Sets or Representations indicated as being compliant with multiple profiles (but none of those required to be supported by the present document) where some of the profiles are required to be ignored by the rules in this clause and others are not required to be ignored by those same rules are not required to be ignored.
- Where the MPD@profiles attribute includes "urn:hbbtv:dash:profile:isoff-live:2012" as well as some other profile, AdaptationSets and Representations not inferred to have a @profiles attribute that includes "urn:hbbtv:dash:profile:isoff-live:2012" shall be ignored by HbbTV terminals that support only the DASH profile defined in the present document.