

ETSI TS 102 816 V1.1.1 (2007-09)

Technical Specification

Digital Video Broadcasting (DVB); Personal Video Recorder (PVR)/Personal Data Recorder (PDR) Extension to the Multimedia Home Platform

European Broadcasting Union



Union Européenne de Radio-Télévision



Reference

DTS/JTC-DVB-176

Keywords

broadcasting, digital, DVB, TV, video

ETSI

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

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

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

Important notice

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

http://portal.etsi.org/chaicor/ETSI_support.asp

Copyright Notification

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

© European Telecommunications Standards Institute 2007.

© European Broadcasting Union 2007.

All rights reserved.

DECTTM, **PLUGTESTS**TM and **UMTS**TM are Trade Marks of ETSI registered for the benefit of its Members.
TIPHONTM and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members.
3GPPTM is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Contents

Intellectual Property Rights	5
Foreword.....	5
Introduction	5
1 Scope	7
2 References	7
3 Definitions and abbreviations.....	8
3.1 Definitions	8
3.2 Abbreviations	8
4 Conventions.....	9
5 Basic Architecture	9
5.1 Relationship with MHP and GEM Specifications	9
5.2 Overview (informative).....	9
5.3 General requirements	11
6 Recording and playback process	11
6.1 Managing scheduled recording.....	11
6.2 The recording process	12
6.2.1 Identifying streams to be recorded.....	12
6.2.2 Identifying and recording MHP applications.....	12
6.3 Managing completed recordings	13
6.4 Playback	13
6.5 Timeshift	14
6.6 TV-Anytime	14
7 Metadata	15
7.1 TV-Anytime	15
7.1.1 Definition.....	15
7.1.2 Transport by broadcast channel	15
7.1.3 Transport by interaction channel	15
8 Application model.....	15
8.1 Playback of recorded applications.....	15
8.2 Service contexts and support for virtual channels	15
8.3 Resource Management	16
8.4 Modifications to MHP 1.0 application model specification	16
9 Application signalling	16
9.1 Recording specific security attributes.....	16
9.1.1 Signalling.....	16
9.1.2 Determining broadcaster permissions	18
9.2 Signalling for application recording	19
9.2.1 Application recording descriptor	19
9.3 Extensions to application signalling	19
9.4 Modifications to MHP 1.0 application signalling specification	19
10 DVB-J platform.....	20
10.1 PDR.....	20
10.1.1 Common Core.....	20
10.1.2 DVB Extensions	22
10.1.3 Related Content	22
10.2 TV-Anytime	22
10.2.1 Content referencing.....	22
10.2.2 Metadata	22
10.3 Integration between PDR and TV-Anytime	23
10.3.1 TV-Anytime based recording	23

10.3.2	Content Identification API.....	23
10.4	Version properties	23
10.5	Extended semantics for MHP DVB-J Platform.....	24
10.5.1	User Settings and Preferences API	24
10.5.2	DVB Service Information API.....	24
10.5.3	Application discovery and launching APIs.....	24
10.6	Modifications to MHP 1.0 DVB-J Platform.....	25
11	Security.....	26
11.1	Permission Request File Extensions.....	26
12	System integration.....	26
12.1	TV-Anytime content referencing	26
12.1.1	Broadcast channel usage	27
12.1.2	Resolution by interaction channel.....	27
13	Detailed platform profile definitions	27
14	Registry of constants	28
14.1	System constants	28
15	Minimum Platform Capabilities.....	28
	Annex A (informative): Responsibilities of GEM Recording Specifications.....	29
A.1	Required responsibilities	29
A.2	Optional responsibilities.....	30
	Annex B (informative): Bibliography.....	31
	History	32

Intellectual Property Rights

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

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

Foreword

This Technical Specification (TS) has been produced by Joint Technical Committee (JTC) Broadcast of the European Broadcasting Union (EBU), Comité Européen de Normalisation ELECTrotechnique (CENELEC) and the European Telecommunications Standards Institute (ETSI).

NOTE: The EBU/ETSI JTC Broadcast was established in 1990 to co-ordinate the drafting of standards in the specific field of broadcasting and related fields. Since 1995 the JTC Broadcast became a tripartite body by including in the Memorandum of Understanding also CENELEC, which is responsible for the standardization of radio and television receivers. The EBU is a professional association of broadcasting organizations whose work includes the co-ordination of its members' activities in the technical, legal, programme-making and programme-exchange domains. The EBU has active members in about 60 countries in the European broadcasting area; its headquarters is in Geneva.

European Broadcasting Union
CH-1218 GRAND SACONNEX (Geneva)
Switzerland
Tel: +41 22 717 21 11
Fax: +41 22 717 24 81

Founded in September 1993, the DVB Project is a market-led consortium of public and private sector organizations in the television industry. Its aim is to establish the framework for the introduction of MPEG-2 based digital television services. Now comprising over 200 organizations from more than 25 countries around the world, DVB fosters market-led systems, which meet the real needs, and economic circumstances, of the consumer electronics and the broadcast industry.

Introduction

The specifications for the following Java packages are contained in archive ts_102816v010101p0.zip which accompanies the present document:

- org.dvb.media.tvanytime
- org.dvb.media.rct
- org.dvb.pvr
- org.dvb.pvr.navigation
- org.dvb.si.tva
- org.dvb.tvanytime.metadata
- org.dvb.tvanytime.resolution
- org.dvb.tvanytime.pvr

- org.dvb.tvanytime.pvr.navigation
- org.dvb.xml.jdom
- org.dvb.locator.content

This is the first public release of the PVR/PDR extension to MHP. The aim of the present document is to encourage implementations of:

- Receivers and middleware.
- Applications.
- Conformance tests.

DVB welcomes feedback from the developers of these implementations. Past experience suggests that this feedback will result in a revised version of the present document and that the first release of conformance tests for the PVR/PDR extension to MHP will address such a revision.

1 Scope

The present document defines the extension to the MHP specification supporting the recording of digital television content. It includes the scheduling of recordings, the management of scheduled recordings, the playback of completed recordings and the management of completed recordings. It includes the recording and playback of MHP applications that form part of the digital television content being recorded. It includes access to the metadata and content resolution mechanisms defined by TV-Anytime.

2 References

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

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <http://docbox.etsi.org/Reference>.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

- [1] ETSI ES 201 812: "Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.0.3".
- [2] ETSI TS 102 812: "Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.1.1".
- [3] ETSI TS 102 822 (all parts): "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime Phase 1")".
- [4] ETSI TS 102 323 (V1.2.1): "Digital Video Broadcasting (DVB); Carriage and signalling of TV-Anytime information in DVB transport streams".
- [5] ETSI TS 102 823 (V1.1.1): "Digital Video Broadcasting (DVB); Specification for the carriage of synchronized auxiliary data in DVB transport streams".
- [6] ETSI TS 102 817 (V1.1.1): "Digital Video Broadcasting (DVB); Digital Recording Extension to Globally Executable Multimedia Home Platform (GEM)".
- [7] ETSI EN 300 468 (V1.5.1): "Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 102 817 [6] and the following apply:

complete piece of content: content between two DVB-SI event boundaries

content recording and access policy: policy controlling the ability for MHP applications from one broadcaster to use content from a different broadcaster

MHP 1.0: generic term for ES 201 812 [1] and its predecessors when known as TS 102 812 [2]

MHP 1.1: generic term for TS 102 812 [2]

MHP-PVR terminal: MHP terminal additionally conforming to all mandatory requirements of the present document

recordable application: MHP application which is signalled as to be recorded along with a piece of TV content and which does not rely on dynamic data or on synchronization with audio/video

timeshift: simultaneous recording and playback of digital television content such that the playback can be paused while recording continues hence enabling playback to continue at a later time with the possibility that none of the content has been lost

timeshift buffer: the buffer on the storage medium of the MHP-PVR terminal that is used for when timeshift behaviour is in operation

NOTE: The present document is intentionally silent about whether this is a fixed size buffer (e.g. 5 minutes) or a variable size buffer (e.g. all free space on the device).

transport keys: well known VCR control keys (play, pause, fast forward, fast rewind, etc.)

NOTE: These never go directly to MHP applications and are exclusively reserved for any manufacturer PVR/PDR user interface within the navigator.

tva_id: the "TV-Anytime event identifier" defined by TS 102 323 [4]

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AIT Application Information Table

NOTE: As defined in clause 10.4 of ES 201 812 [1] and TS 102 812 [2].

API Application Program Interface

NOTE: Sometimes also referred to as Application Programming Interface.

BNF Backus-Naur Form

CRID Content Reference IDentifier

NOTE: As introduced by clause 5 of TS 102 822-2 [3] and defined in clause 8 of TS 102 822-4 [3].

DAVIC Digital Audio Visual Council

DSM-CC Digital Storage Media - Command and Control

DVB Digital Video Broadcasting

DVB-SI Digital Video Broadcasting - Service Information

NOTE: As defined in EN 300 468 [7].

DVR Digital Video Recorder

EIT Event Information Table

MHP Multimedia Home Platform

NOTE: As defined in either ES 201 812 [1] or TS 102 812 [2].

PDR	Personal Digital Recorder
PVR	Personal Video Recorder
SI	Service Information
STD	Service Description Table

4 Conventions

Where modifications to other documents are defined by quoting text from those other documents and describing how the present document considers that text to be changed, text that the present document adds into the original document is shown underlined and text that the present document removes from the original document is shown with strike-through.

The present document uses the terminology that something "shall be recorded", "should be recorded", "shall not be recorded" or "should not be recorded". This is a convention for the sake of brevity and in all cases, implementations are allowed to record items that "shall not be recorded" and discard them during playback. The term "shall not be recorded" is simply shorter than "shall not be recorded, or if recorded, shall be discarded during playback" and the latter is what is meant.

5 Basic Architecture

5.1 Relationship with MHP and GEM Specifications

Implementations of the present document shall additionally implement all mandatory requirements of either ES 201 812 [1] or TS 102 812 [2].

NOTE: Future versions of the present document may remove ES 201 812 [1] from the above and require implementation of all mandatory requirements of TS 102 812 [2].

All normative clauses of TS 102 817 [6] shall apply regardless of whether the clause is explicitly mentioned below.

5.2 Overview (informative)

Figure 1 shows an overview of the architecture assumed by the present document. A number of aspects are omitted for the sake of clarity.

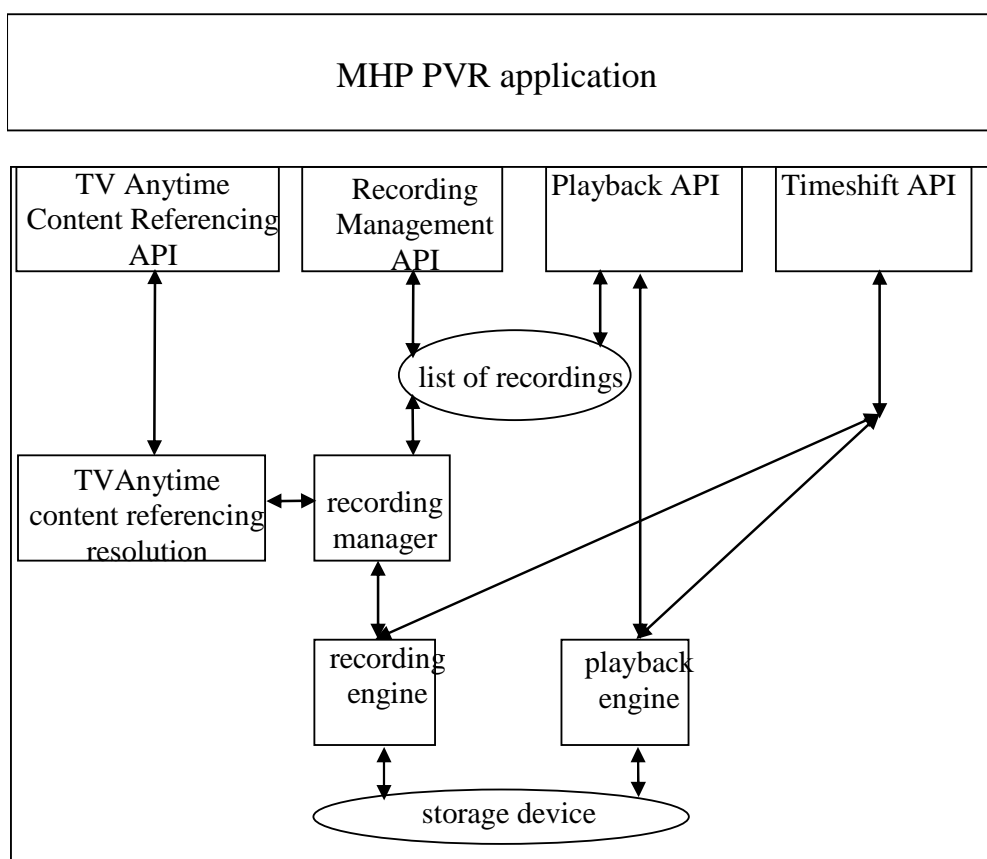


Figure 1: Simplified Architecture of MHP-PVR

In figure 1, the core of the architecture is the list of recordings and the recording manager. The list of recordings includes both pending and completed recordings. Pending recordings may include ones fixed in terms of time and duration on a specific TV channel as well as ones which can move in either time or channel - e.g. ones defined by DVB-SI or by the TV-Anytime CRID. Among other things, the recording manager is responsible for:

- managing pending recordings;
- interfacing to the recording engine when recordings are to be performed;
- interfacing to an implementation of the TV-Anytime content referencing mechanism for pending recordings defined by a TV-Anytime CRID;
- updating the status of recordings in the recording list as they are made.

Some pending recordings may be for more than one item of content, (e.g. a series identified by a CRID) in which case the list of recordings grows when a recording is made as the completed recording is added to the list of recordings. The original series recording request remains in the list as still pending until all elements of the list have been recorded.

MHP-PVR applications can use the recording management API to:

- request recordings to be made;
- to manage pending recordings;
- to search for pending or completed recordings which match certain criteria.

MHP-PVR applications can use the TV-Anytime content referencing API to access the implementation of the TV-Anytime content referencing mechanism. Pausing of the current "live" TV channel is supported via the timeshift API. Playback of completed recordings is supported via extended versions of the existing MHP media playback APIs.

Items that have been omitted for clarity in the above description include:

- a user interface to the recording manager provided by the receiver manufacturer;
- a database for TV-Anytime defined metadata and an API to access that database;
- mechanisms to enable accurate recording techniques to use by the recording engine.

5.3 General requirements

The following requirements of ES 201 812 [1] shall also apply to the packages defined in the present document:

- Clause 11.2.7 "Event model in DAVIC and DVB APIs".
- Clause 11.2.5 "Event listeners".
- Applications shall not define classes or interfaces in any package namespace defined in the present document. MHP terminals shall enforce this using the `SecurityManager.checkPackageDefinition` mechanism.

6 Recording and playback process

6.1 Managing scheduled recording

In addition to the activities defined in clause 6.1 of TS 102 817 [6], the process for managing scheduled recordings shall include the activities:

- 1) cross referencing the set of pending recordings with scheduling information to identify changes in the schedule and to schedule recordings not previously scheduled which can now be scheduled;
- 2) resolving conflicts between individual pending recordings regardless of whether the recording originated via the API(s) defined in the present document, via a user interface provided by the manufacturer of the MHP-PVR terminal or even via an in-home network;

NOTE: The present document intentionally does not define any specific algorithms or mechanisms for resolving these conflicts. The only requirement is that implementations include such a mechanism.

- 3) optionally discarding recording requests which are still in a pending state once their validity period has expired (but not before this);
- 4) optionally discarding failed recording requests once their validity period has expired (but not before this).

The handling of multiple requests to record the same piece of content is implementation specific. Some implementations may treat this as a conflict and resolve it via whatever mechanism they provide for resolving conflicts. Other implementations may logically merge the recording requests also merging the recording properties if they differ.

It is implementation dependent whether attempts are made to update the following information between when a recording request is first scheduled and when the recording process starts:

- 1) broadcaster permissions for any recording requests marked as not having been checked for broadcaster permissions (see clause 9.1.2 Determining broadcaster permissions);
- 2) DVB-SI descriptive metadata associated with the recording request.

6.2 The recording process

In addition to the activities defined in clause 6.2 of TS 102 817 [6], the recording process shall include the following activities:

- 1) Performing accurate recording using the mechanisms defined in clause 11 of TS 102 323 [4].
 - 2) Acquiring the broadcaster permissions for the pieces of content in the recording as defined in clause 9.1.2 Determining broadcaster permissions. If evaluating the broadcaster permissions according to clause 10.1.1 Common Core determines that permission for recording is to be denied (see "Make a scheduled record request"), the recording shall not be started and the recording request shall transition to the FAILED_STATE.
 - 3) Associating with the recording the DVB-SI descriptive metadata (title, synopsis & extended event descriptor) for the piece of content which makes up the largest proportion of the recording and that for all complete pieces of content found in the recording. Any descriptive metadata previously associated with the recording request is discarded.
 - 4) Associating broadcaster permission descriptors with recordings as follows:
 - for recordings defined by DVB-SI event_id, (both with and without time and duration being specified), the broadcaster permission descriptor for the specified DVB-SI event_id shall be stored;
 - for recordings defined only by time and duration (without tva_id or event_id), the broadcaster permission descriptors for all DVB-SI events in that interval shall be stored.
- NOTE: Recordings defined by time, duration (without either tva_id or event_id) are intended to be used where EIT present/following are very out of step with the content. In this case, broadcaster permissions descriptor changes will also be (very) out of step with the content to which they refer.
- 5) Generating segments for each DVB-SI event in a recording request that is specified solely by time and duration and which spans more than one complete piece of content as determined by DVB-SI event boundaries.

The protocols for transmitted timelines referred to by TS 102 817 [6] shall be both the synchronized auxiliary data as defined in TS 102 823 [5] and DSMCC normal play time as defined in ES 201 812 [1] and TS 102 812 [2].

6.2.1 Identifying streams to be recorded

In the present document, the term "recordable streams" used in TS 102 817 [6] shall be interpreted to mean those streams defined in clause 7.2 of ES 201 812 [1] and TS 102 812 [2].

Identifying the streams to be recorded shall be done as follows:

- for each type of recordable stream, if the number of streams of each type present is less than or equal to the limits in the recording capability of the MHP-PVR terminal then all the streams of that type shall be recorded;

NOTE: Minimum capabilities for recording streams are defined in clause 15 of the present document.

- where more streams of any one type exist than the MHP-PVR terminal can record, the decision on what to record shall be according to clause 11.4.2.3 of ES 201 812 [1] and TS 102 812 [2].

6.2.2 Identifying and recording MHP applications

Identifying recordable MHP applications shall be done as follows:

- If an application does not rely on dynamic data and does not rely on synchronization with audio/video and if this application is signalled as to be recorded (the scheduled_recording_flag in the application recording descriptor is set to '1'), then the application shall be recorded.
- If an MHP-PVR terminal is able to reconstruct during playback the timing of the delivery of dynamic data then it should record applications that rely on this as long as they do not rely on other features the MHP-PVR terminal does not support.

- If an MHP-PVR terminal is able to reconstruct during playback timing of synchronization to audio/video then it should record applications that rely on this as long as they do not rely on other features the MHP-PVR terminal does not support.
- In all other cases, recording that application is not required. This includes MHP applications without specific recordability signalling (i.e. without an application recording descriptor in their AIT).

NOTE 1: Implementations are however free to attempt to record more applications than are required and play them back. Substantial care is needed to offer the end-user the experience that was intended by the developer of the original program.

During the recording process, the MHP-PVR terminal shall:

- 1) monitor for changes in the AIT or AITs as defined in clause 10.4.2 "AIT transmission and monitoring" of ES 201 812 [1];
- 2) capture all AITs which are detected by this monitoring process;
- 3) identify all recordable applications listed in that AIT and start capturing those applications and associated streams as defined by the application recording descriptor.

NOTE 2: It is implementation dependent when the MHP-PVR terminal captures the application.

6.3 Managing completed recordings

In addition to the activities defined in clause 6.3 of TS 102 817 [6], the process for managing completed recordings shall include the following activities:

- 1) Deleting the recording at some point once the expiration period is past. There is no requirement for this to be accurately enforced, either by deleting the recording or by making it inaccessible through the API.
- 2) Maintaining with all completed recordings the following information:
 - All successfully captured AITs, applications and associated streams except for those associated with pieces of content which have not been completely recorded and which do not form the largest proportion of the recording. The retention of these is implementation dependent.
 - The descriptive metadata associated with pieces of content in the recording as defined above.
 - Any access rights which that broadcaster defined for applications not coming from them.

6.4 Playback

The process for playback shall be as defined in clause 6.4 of TS 102 817 [6].

During the playback of content recorded as the result of a scheduled recording, the following behaviour shall be supported.

Table 1: Events during normal playback and resulting behaviour

Event	Behaviour	Result on screen	Java Event
Fast forward to end of stream when recording is in progress	End of media event generated to any registered applications	Playback continues at rate 1.0 at the end of the stream	EndOfContentEvent
Rewind to beginning of stream	Switch to pause mode	First frame frozen	org.ocap.shared.media.BeginningOfContentEvent
Fast forward to end of stream when recording is not in progress and play to end of stream	End of media generated to any registered applications	Last frame frozen	EndOfMediaEvent

6.5 Timeshift

The process for timeshift recording and playback shall be as defined in clause 6.5 of TS 102 817 [6] with the following clarification:

- 1) All streams within the service shall be recorded including those carrying dynamic data such as MHP application signalling, DSMCC object carousel(s), stream events and MPEG-2 private sections to be accessed via the section filtering API.
- 2) All applications within the service are considered to be recordable unless explicitly excluded by having the `time_shift_flag` in their application recording descriptor set to '0'.
- 3) During playback, all recorded streams shall be decoded as if a live broadcast was being decoded, and the timing of dynamic data shall be reconstructed.
- 4) The MHP-PVR terminal shall associate a timeshift buffer with the service context created by the MHP navigator in which the user interface of the navigator selects services. The present document does not define mechanisms to associate timeshift buffers with other service contexts.

6.6 TV-Anytime

In a TV-Anytime environment, the following additional activities are required as part of the "managing scheduled recording" process defined in clause 6.1 Managing scheduled recording above:

- 1) resolving requests to record group CRIDs into their constituent elements.
- 2) monitoring when an additional resolution information becomes available for incompletely resolved CRIDs and acting on that additional information.

In a TV-Anytime environment, the following additional activities are required as part of the "recording" process defined in clause 6.2 The recording process above:

- 1) Associating with completed recordings all CRIDs that were signalled with the content at time of recording using the content identifier descriptor defined in clause 12 of TS 102 323 [4].
- 2) Associating the corresponding TV-Anytime metadata with recordings where any of the following pieces of content have CRIDs signalled with the content identifier descriptor:
 - the piece of content which makes up the largest proportion of the recording;
 - all complete pieces of content found in the recording.

The corresponding TV-Anytime metadata for a CRID is defined as follows. For a group CRID, it is the GroupInformation. For a leaf CRID, it is the ProgramInformation and segmentation information if signalled. If instance metadata is available for the selected instance, it over-rides equivalent fields in the ProgramInformation.

Segmentation information shall always be acquired at the end of a recording so as to include dynamic information. Other metadata may be acquired at any point during the recording.

If a metadata database was specified when the recording was specified, the metadata shall be obtained from that database. Otherwise the database to obtain the metadata is implementation dependent with the constraint that databases in the transport stream carrying the content to be recorded shall be used if they exist.
- 3) For recordings defined by `tva_id`, the broadcaster permission descriptors for all DVB-SI events in scope of the specified `tva_id` shall be stored. Where the content to be recorded spans more than one DVB-SI event_id, the LeafRecordingRequest shall have segments for the piece of content which makes up the largest proportion of the recording and for all complete pieces of content in the recording.

7 Metadata

7.1 TV-Anytime

NOTE: The present document does not define functional profiles or levels of the TV-Anytime specifications. One place where such activity is happening in the DTG TV-Anytime Test Bed project, see Bibliography.

7.1.1 Definition

The metadata defined by clause 8 of TS 102 323 [4] shall be supported.

7.1.2 Transport by broadcast channel

Metadata transport as defined by clause 9 of TS 102 323 [4] shall be supported.

TV-Anytime metadata fragments obtained from a DVBDatabase shall include `fragmentVersion` and `fragmentId` attributes values which are obtained from the corresponding fields of the encapsulation structure defined in TS 102 822-3-2 [3]. Any existing values found in the fragments will be discarded and replaced.

- The 24-bit unsigned integer `fragmentId` field shall be represented by a hexadecimal string in the XML as would be accepted by `java.lang.Integer.parseInt(String s, 16)`.
- The 8-bit unsigned integer `fragmentVersion` field shall be represented by a long value in the XML.

7.1.3 Transport by interaction channel

Metadata transport as defined by of TS 102 822-6 [3] shall be supported.

8 Application model

8.1 Playback of recorded applications

Clause 9 of TS 102 817 [6] shall be supported and additionally constrained as follows:

- When playback leaves trick-mode and returns to normal, MHP-PVR terminals may delay re-starting applications for up to one minute. The behaviour for this minute is implementation dependent.

8.2 Service contexts and support for virtual channels

MHP-PVR terminals shall be able support at least the following service contexts simultaneously:

- a) the service context created by the MHP terminal implementation that in MHP 1.0 is used by the navigator to present broadcast services;
- b) a service context created by MHP-PVR applications which could be used for the presentation of virtual channels.

Both of these can be used by MHP-PVR applications for the presentation of both broadcast and recorded content.

Simultaneous support for these service contexts shall include support for executing MHP applications simultaneously in both service contexts. This shall include two instances of the same MHP application should an application running in the service context for broadcast applications happen to be started as a result of the playback of recorded content.

NOTE: The intended use case for the above is virtual channels where the application controlling the virtual channel is running in the service context for normal broadcast applications and is selecting the content making up the virtual channel in a second service context which it has created.

8.3 Resource Management

The existing language in ES 201 812 [1] or TS 102 812 [2] clause 11.2.9 "Intra application media resource management" shall be extended to address the situation where an explicit request for media decoding resources by an application conflicts with existing resource usage associated with the service context in which that application is running. Specifically, if an MHP application requests the presentation of audio and/or video content and this needs to re-use the video and audio decoders used to present the currently selected service in the service context in which that application is running, those decoders shall be used for the newly requested presentation and shall no longer be used to present the video and audio of the currently selected service. Hence if an MHP application running in one service context creates another service context and then selects a service containing video and audio in that other service context, the selection in the second service context shall, if necessary, take media decoding resources away from any service being presented in the first service context.

8.4 Modifications to MHP 1.0 application model specification

When the present document is used in devices supporting ES 201 812 [1], the following changes shall be assumed to the text found in that document:

- a) In clause 9.1.6, the following text:

"only a single application with a given Application identification is allowed to run at any time"

shall be assumed to be changed as follows:

*"only a single application with a given Application identification is allowed to run at any time **in a single service context**".*

9 Application signalling

9.1 Recording specific security attributes

9.1.1 Signalling

The following descriptor is defined in order to enable one broadcaster or operator to signal the rights which they wish to grant (or exclude) to MHP applications from other organizations. This descriptor is defined for use in the SDT and the EIT. When used in the SDT, the scope of this descriptor is that service. When used in the EIT, the scope of this descriptor is the event concerned.

Table 2: Broadcaster permission descriptor syntax

	No. of Bits	Identifier	Value
broadcaster_permission_descriptor() {			
descriptor_tag	8	uimsbf	0x7B
descriptor_length	8	uimsbf	
this_organization_loop_length	8	uimsbf	
For(i=0;i<N;i++) {			
this_organization_id	32	bslbf	
}			
For(i=0; i<N; i++) {			
other_organization_id	32	bslbf	
schedule_application_initiated_recording	1	bslbf	
read_pending_application_initiated_recording	1	bslbf	
modify_application_initiated_recording	1	bslbf	
delete_application_initiated_recording	1	bslbf	
read_metadata	1	bslbf	
read_user_initiated_recordings	1	bslbf	
read_completed_application_initiated_recordings	1	bslbf	
user_initiated_record_now	1	bslbf	
application_initiated_record_now	1	bslbf	
play_user_initiated	1	bslbf	
play_application_initiated	1	bslbf	
preview_user_initiated	1	bslbf	
preview_application_initiated	1	bslbf	
delete_user_initiated_recordings	1	bslbf	
delete_application_initiated_recordings	1	bslbf	
reserved_future_use	1	bslbf	
}			
}			

descriptor_tag: This 8 bit integer with value 0x7B identifies this descriptor.

this_organization_loop_length: This 8-bit field gives the total length in bytes of the following loop.

this_organization_id: This field identifies the MHP organization_id or ids of this broadcaster, i.e. the one providing the content in the scope of the descriptor.

other_organization_id: The MHP organization_id of another broadcaster whose MHP applications this broadcaster wishes to grant or deny access to the content to which this descriptor refers. The value of '0' is a wildcard meaning all organization_ids except those explicitly listed in this loop and except for this_organization_id. If the value of the this_organization_id field is explicitly listed, the fields associated with it shall be ignored.

The following 1 bit fields are flags. When set to 1, the broadcaster identified by this_organization_id permits (or excludes) MHP applications from the broadcaster identified by other_organization_id from performing the identified operation on content within the scope of this descriptor.

Table 3: Broadcaster permission descriptor field semantics

Field	Operation	Meaning of flag when set to '1'
schedule_application_initiated_recording	scheduling recordings	permit
read_pending_application_initiated_recording	reading previously scheduled application initiated recordings	permit
modify_application_initiated_recording	modify previously scheduled application initiated recordings	permit
delete_application_initiated_recording	delete previously scheduled application initiated recordings	permit
read_metadata	read the metadata stored with a piece of content recorded through application initiated recording	permit
read_user_initiated_recordings	read references to content recorded in user initiated recordings from the list of recorded content	exclude
read_completed_application_initiated_recordings	read references to content recorded in application initiated recordings from the list of recorded content	exclude
user_initiated_record_now	record currently available content via user initiated recording	permit
application_initiated_record_now	record currently available content via application initiated recording	permit
play_user_initiated	play content which was recorded through user initiated recording	exclude
play_application_initiated	play content which was recorded through application initiated recording	exclude
preview_user_initiated	preview content which was recorded through user initiated recording	exclude
preview_application_initiated	preview content which was recorded through application initiated recording	exclude
delete_user_initiated_recordings	delete content which was recorded through user initiated recording	exclude
delete_application_initiated_recordings	delete content which was recorded through application initiated recording	exclude

The default for content not in the scope of one of these descriptors shall be equivalent to a descriptor with the other_organization_id field being zero and all the following fields being zero.

9.1.2 Determining broadcaster permissions

The process for determining the broadcaster permissions for a piece of content to be recorded is as follows:

- 1) Determine if the SDT of the service containing the content to be recorded is available (without tuning), if not mark the recording request as not having been checked for broadcaster permissions and continue with step 7.
- 2) Determine if EIT information is available (without tuning) for the content to be recorded, if not mark the recording request as not having been checked for broadcaster permissions and continue with step 7.
- 3) If no DVB-SI event_id was specified in the recording request, determine one from the EIT information based on the time and duration specified in the recording request.
- 4) Based on the event_id, determine if a broadcaster permission descriptor for this content is present or absent in the EIT, if one is present, use this definition of the permissions and continue with step 7.
- 5) If a broadcaster permission descriptor is absent in the EIT, check for one in the SDT of the service for the content to be recorded, if one is present, use this definition of the permissions and continue with step 7.
- 6) If there is no broadcaster permission descriptor present in the SDT, use the default permissions.
- 7) End of determination process.

In the above process, if the piece of content to be recorded is listed in the EIT present/following table then the references to EIT above shall refer to the EIT present/following. Otherwise the references to EIT above shall refer to the EIT schedule.

Implementations shall attempt to determine the broadcaster permission descriptor for a recording request as follows:

- at the time a recording request is first made (or first resolved in the case of a recording request specified by a TV-Anytime CRID);
- at the time the corresponding recording operation starts and each time a new DVB-SI event starts during the recording. Any changes of broadcaster permission descriptor during the duration of an event shall be ignored.

The most recently determined broadcaster permissions shall apply in the event of conflicts. This includes conflicts between the broadcaster permissions determined at the time the recording operation starts and ones determined earlier as well as changes in broadcaster permissions when a new DVB-SI event starts during the recording.

If the recording request is marked as not having been checked for broadcaster permissions at the time the request is first made, implementations may repeat the attempt to re-acquire the broadcaster permission descriptor before the corresponding recording operation starts. This is however not required.

9.2 Signalling for application recording

9.2.1 Application recording descriptor

The application signalling defined in clause 8 of TS 102 817 [6] and the application recording descriptor defined in annex A of TS 102 817 [6] shall be supported. For the `av_synced_flag`, trigger events are as defined in clause B.2.4 of ES 201 812 [1] and TS 102 812 [2].

9.3 Extensions to application signalling

The present document extends the application control codes defined in clauses 10.6.2.1 and 10.6.2.2 of ES 201 812 [1] and TS 102 812 [2] as follows:

Table 4: Extensions to application control codes

Code	Identifier	Semantics
0x07	PLAYBACK_AUTOSTART	Application shall not be run either direct from broadcast or when in timeshift mode. When a recording is being played back from storage, the application shall be presented as if it was AUTOSTART.

9.4 Modifications to MHP 1.0 application signalling specification

When the present document is used in devices supporting ES 201 812 [1], the following changes shall be assumed to the text found in that document:

- a) In clause 10.5.2, the following text:

"Only a single instance of an application with a particular application identifier is allowed to execute at any time. So, if an application is already running then another instance of the same application shall not be launched."

shall be assumed to be changed as follows:

"Only a single instance of an application with a particular application identifier is allowed to execute at any time in the same service context. So, if an application is already running in a service context then another instance of the same application shall not be launched in that same service context."

10 DVB-J platform

10.1 PDR

10.1.1 Common Core

Clause 7 of TS 102 817 [6] shall be supported.

The behaviour of methods depending on recording request specific security attributes is defined as follows:

- 1) Where the application calling the method is from the same organization as the broadcaster of the content addressed by the recording request (as defined by the `this_organization_id` field in the broadcaster permission descriptor), there shall be no restrictions. No method shall throw `org.ocap.shared.dvr.AccessDeniedException` under these circumstances. Recording requests for that content shall always be visible to such applications.
- 2) Where the application calling the method is from a different organization from the broadcaster of the content addressed by the recording request, the restrictions shall be as defined in table 5.

Table 5: Recording specific security attribute policy for content from one broadcaster and applications from another broadcaster

Operation and corresponding method call or calls	User initiated action	Application initiated action
Requests for recordings		
Make a scheduled record request RecordingManager.record() (note 2)	Yes	No, unless permitted by the schedule_application_initiated_recording flag
Read a scheduled record request (note 4) RecordingManager.getEntries() RecordingManager.getEntries(RecordingListFilter) RecordingManager.addRecordingChangedListener(RecordingChangedListener) CRIDRecordingManager.getCRIDRecordingRequest(CRID)	Yes	No, unless permitted by the read_pending_application_initiated_recording flag (note 1)
Modify a scheduled record request RecordingRequest.setRecordingProperties(RecordingSpec) LeafRecordingRequest.stop() RecordingRequest.addAppData(int,java.io.Serializable) RecordingRequest.removeAppData(int)	Yes	No, unless permitted by the modify_application_initiated_recording flag (note 1)
Delete a scheduled record request LeafRecordingRequest.cancel() ParentRecordingRequest.cancel()	Yes	No, unless permitted by the delete_application_initiated_recording flag (note 1)
Information about recordings already made		
Reading content metadata stored with a piece of content DVBRecordedService.getProgramEvents() DVBRecordedService.getCRIDs() DVBLEafRecordingRequest.getProgramEvents()	Yes	Yes, unless excluded by the read_metadata flag
Writing/Modifying mutable metadata stored with a piece of content RecordedService.setMediaTime	No	No
Access to a list of all content recorded (note 5) RecordingManager.getEntries() RecordingManager.getEntries(RecordingListFilter) RecordingManager.addRecordingChangedListener(RecordingChangedListener)	Yes, unless excluded by the read_user_initiated_recordings flag	Yes, unless excluded by the read_completed_application_initiated_recordings flag
Recordings of content		
Record content now RecordingManager.record() (note 3)	No, unless permitted by the user_initiated_record_now flag	No, unless permitted by the application_initiated_record_now flag
Play recorded content LeafRecordingRequest.getService()	Yes, unless excluded by the play_user_initiated flag	Yes, unless excluded by the play_application_initiated flag
Preview recorded content No applicable method in the present document	Yes, unless excluded by the preview_user_initiated flag	Yes, unless excluded by the preview_application_initiated flag
Delete Content RecordingRequest.delete()	Yes, unless excluded by the delete_user_initiated_recordings flag (note 1)	Yes, unless excluded by the delete_application_initiated_recordings flag (note 1)
NOTE 1: Applications from the same organization as the one that originally made a scheduled recording request are allowed to read, modify and delete that recording request regardless of the signalling in the broadcaster permission descriptor.		
NOTE 2: Applies to all sub-classes of RecordingSpec except for ServiceContextRecordingSpec.		
NOTE 3: Applies only to ServiceContextRecordingSpec.		
NOTE 4: Applies to all ParentRecordingRequests and LeafRecordingRequests in the following states - FAILED_STATE,PENDING_NO_CONFLICT_STATE, PENDING_WITH_CONFLICT_STATE.		
NOTE 5: Applies to LeafRecordingRequests in the following states COMPLETED_STATE, IN_PROGRESS_INSUFFICIENT_SPACE_STATE, IN_PROGRESS_STATE, INCOMPLETE_STATE.		

The restrictions defined in this table manifest themselves in the specifications as any of the following:

- the throwing of `AccessDeniedException` for methods defined to throw that exception;
 - `RecordingChangedEvents` not being sent to a `RecordingChangedListener`;
 - `RecordingRequests` not being returned by `RecordingManager.getEntries`.
- 3) User initiated recordings made by the end-user using the user interface of the navigator shall be visible to MHP applications as user initiated recordings and the normal policy governing the visibility of these shall apply. It is implementation dependent whether any application initiated recordings made by the navigator (i.e. without involving the end-user) are visible to MHP applications.

If a recording includes both timelines as defined by DSMCC Normal Play Time and timelines as defined by TS 102 823 [5] then instances of `TimeLine` shall be returned for the timelines defined by the latter and not the former.

10.1.2 DVB Extensions

The `org.dvb.pvr` and `org.dvb.pvr.navigation` packages shall be supported.

All instances of `org.ocap.shared.dvr.RecordedService` created by the MHP-PVR terminal shall be instances of `org.dvb.pvr.DVBRecordedService`.

All instances of `org.ocap.shared.dvr.LeafRecordingRequest` created by the MHP-PVR terminal shall be instances of `org.dvb.pvr.DVBLeafRecordingRequest`.

The signalling for CRIDs in the method `DVBRecordedService.getCRIDs` shall be the content identifier descriptor as defined in clause 12.1 of TS 102 323 [4].

Implementations shall provide a mechanism to separate user and application initiated recordings as identified by the `userInitiated` parameter to the constructor of the `DVBRecordingProperties` class. This mechanism shall ensure that applications do not abuse the user-initiated recording mechanism to make what are in fact application initiated recordings. The present document does not require any specific mechanism. One example of a mechanism would be for the implementation to show a user interface dialogue to the end-user for them to explicitly approve (or not) each user initiated recording but not to show such a dialogue for application initiated recordings. Where it is necessary to distinguish between a user-initiated action and an application-initiated action (e.g. an application attempts to delete a user-initiated recording) a similar mechanism is required.

10.1.3 Related Content

The `org.dvb.media.rct` package shall be supported. This shall map onto the promotional links mechanism that is defined in clause 10 of TS 102 323 [4].

10.2 TV-Anytime

10.2.1 Content referencing

The `org.dvb.tvanytime.resolution` and `org.dvb.locator,content` packages shall be supported.

When the method `org.dvb.tvanytime.resolution.ResolutionResponse.getChildren` returns a vector of `ContentLocation` objects, references to DVB locators in the resolution result shall be represented by instances of `org.dvb.pvr.DVBContentLocation`.

10.2.2 Metadata

This `org.dvb.tvanytime.metadata` and `org.dvb.xml.jdom` packages shall be supported.

The classification schemes defined in annex A of TS 102 822-3-1 (V1.2.1) [3] shall be supported by the platform with the exception of the `ActionTypeCS`. These resident classification schemes can be referenced using the aliases listed in table 6.

Table 6: Aliases for resident classification schemes

Alias	URL
Atmosphere	urn:tva:metadata:cs:AtmosphereCS:2004
AudioPurpose	urn:tva:metadata:cs:AudioPurposeCS:2004
ContentAlert	urn:tva:metadata:cs:ContentAlertCS:2004
Content	urn:tva:metadata:cs:ContentCS:2004
ContentCommercial	urn:tva:metadata:cs:ContentCommercialCS:2002
IPISDNS	urn:dvb:ipisdns:2006
Format	urn:tva:metadata:cs:FormatCS:2004
HowRelated	urn:tva:metadata:cs:HowRelatedCS:2004
IntendedAudience	urn:tva:metadata:cs:IntendedAudienceCS:2004
Intention	urn:tva:metadata:cs:IntentionCS:2004
Language	urn:tva:metadata:cs:LanguageCS:2004
MediaType	urn:tva:metadata:cs:MediaTypeCS:2004
Origination	urn:tva:metadata:cs:OriginationCS:2004
PurchaseType	urn:tva:metadata:cs:PurchaseTypeCS:2004
Role	urn:mpeg:mpeg7:cs:RoleCS:2001
TVARole	urn:tva:metadata:cs:TVARoleCS:2004
UnitType	urn:tva:metadata:cs:UnitTypeCS:2004

Classification Scheme aliases shall be supported by all methods of the `ClassificationScheme` class where a string value is used to reference a classification scheme or a controlled term. In methods which include a database argument, the platform shall first attempt to resolve aliases against the `CSAlias` information provided by the specified database. If this fails it shall attempt to resolve them against the aliases used for the inbuilt Classification Schemes. Aliases shall also be supported where string values are used to reference controlled terms in the `DatabaseQuery` class. The platform shall attempt to resolve these aliases against the `CSAlias` information provided by database to which the query is addressed. If this fails it shall attempt to resolve them against the aliases used for the inbuilt Classification Schemes.

Downloadable classification schemes shall be supported by all methods of the `ClassificationScheme` class that include a `Database` argument.

The `FieldIDDefinitionList.getInstance()` method shall return an instance supporting all the fieldIDs defined in clause 5.1.1.1.2 of TS 102 822-6 [3] and an additional fieldID called "DVBContextPath".

The `CapabilityDescriptionsListener` interface allows applications to be informed of the capabilities of an `IPDatabase`. When the `postCapabilityDescriptions` method is called the `description` argument shall contain a "describe_get_Data_Result" element as defined clause 7.1 of TS 102 822-6 [3].

10.3 Integration between PDR and TV-Anytime

10.3.1 TV-Anytime based recording

The `org.dvb.tvanytime.pvr` and `org.dvb.tvanytime.pvr.navigation` packages shall be supported.

All instances of `org.ocap.shared.pvr.RecordingManager` shall be instances of `org.dvb.tvanytime.pvr.navigation.CRIDRecordingManager`.

10.3.2 Content Identification API

The `org.dvb.si.tva` package shall be supported including both the Explicit and Indirect CRID signalling modes defined in clause 12 of TS 102 323 [4].

10.4 Version properties

The properties listed in the following table shall be included in the property set of the `java.lang.System` class. Thus these properties can be retrieved using `java.lang.System.getProperty()`. Since this API returns a string, numeric return values shall be encoded as defined by `java.lang.Integer.toString(int)`.

Table 7: System properties for version interrogation

Property	Semantics	Possible values	Example
mhp.pvr.version.major	Major version number of the version of the present document supported.	Non-negative integer value	"1"
mhp.pvr.version.minor	Minor version number of the version of the present document supported.	Non-negative integer value	"0"
mhp.pvr.version.micro	Micro version number of the version of the present document supported.	Non-negative integer value	"0"

The values of these properties that shall be returned in implementations of the present document are defined in clause 14.1 System constants.

10.5 Extended semantics for MHP DVB-J Platform

The present document defines the following extended behaviours for the APIs defined in ES 201 812 [1] and TS 102 812 [2].

10.5.1 User Settings and Preferences API

The user settings and preferences API defined in clause 11.9.2 of ES 201 812 [1] and TS 102 812 [2] shall be extended as follows:

- An additional standardized preference shall be defined - "Recording List Access".
- The encoding of this shall be as follows – String whose value is "true" if the end-user allows access to the list of all content recorded in the PDR otherwise "false".

NOTE: The additional preference is not included in the list of those accessible to unsigned applications therefore it is only accessible to signed applications.

10.5.2 DVB Service Information API

The DVB service information API defined in clause 11.6.1 of ES 201 812 [1] and TS 102 812 [2] shall be extended as follows:

- Classes implementing org.dvb.si.SIEvent shall also implement org.dvb.si.tva.ContentIdentifierQuery as defined in the present document.
- If an MHP application running as part of the playback of recorded content tries to access service information, then it shall obtain it from a currently available live broadcast and not from the recording.

10.5.3 Application discovery and launching APIs

The Application discovery and launching APIs defined in clause 11.7.2 of ES 201 812 [1] and TS 102 812 [2] shall be extended as follows:

- The class description of org.dvb.application.AppsDatabase shall be considered to be extended as follows:

Applications whose control code is PLAYBACK_AUTOSTART (as defined in clause 9.3 Extensions to application signalling) shall not be considered as "available" or "currently available" as defined here when they are signalled as part of a service being played either live or in timeshift mode. When part of a service being played from a recording, they shall be considered as "available".

10.6 Modifications to MHP 1.0 DVB-J Platform

When the present document is used in devices supporting ES 201 812 [1] the following changes shall be assumed to the text found in that document:

a) Additional fields and methods shall be added to the org.dvb.io.ixc.IxcRegistry class as follows:

```

/**
 * Definition of the scope for bind or rebind - exported object is only
 * visible to Xlets running within the same service context
 * @since MHP 1.1.2
 */
public static final int SERVICE = 1;

/**
 * Definition of the scope for bind or rebind - exported object is only
 * visible to Xlets within the same DVB-HTML application. Note that an
 * embedded Xlet with a different app ID than its enclosing HTML page is
 * still considered to be the same application as that which contains the enclosing page.
 * @since MHP 1.1.2
 */
public static final int PAGE =2;

/**
 * Definition of the scope for bind or rebind - exported object is visible to
 * any Xlet running within the same MHP terminal subject to requirements of the security model.
 * @since MHP 1.1.2
 */
public static final int GLOBAL=3;

/**
 * Exports an object under a given name in the namespace of
 * an Xlet. The name can be any valid non-null String. No
 * hierarchical namespace exists, e.g. the names "foo" and "bar/../foo"
 * are distinct. If the exporting xlet has been destroyed, this method
 * may fail silently.
 *
 * @since MHP 1.1
 * @param xc The context of the Xlet exporting the object.
 * @param name The name identifying the object.
 * @param obj The object being exported
 * @param scope The scope to which the object is to be exported
 *
 * @exception AlreadyBoundException if this Xlet has previously exported an object
 * under the given name.
 *
 * @exception NullPointerException if xc, name or obj is null
 */
public static void bind(javax.tv.xlet.XletContext xc,
                      String name, Remote obj, int scope)
    throws AlreadyBoundException {
}

/**
 * Rebind the name to a new object in the context of an Xlet;
 * replaces any existing binding.
 * The name can be any valid non-null String. No
 * hierarchical namespace exists, e.g. the names "foo" and "bar/../foo"
 * are distinct. If the exporting xlet has been destroyed, this method
 * may fail silently.
 *
 * @param xc The context of the Xlet that exported the object.
 * @param name The name identifying the object.
 * @param obj The object being exported
 * @param scope The scope to which the object is to be exported
 *
 * @since MHP 1.1
 *
 * @exception NullPointerException if xc, name or obj is null
 */
public static void rebind(javax.tv.xlet.XletContext xc,
                        String name, Remote obj, int scope) {
}

```

b) The following text shall be considered to be added to the method `bind(javax.tv.xlet.XletContext, String, Remote)`:

- * <p>The object shall be made visible to other applications in the same service context. A call
- * to `bind(xc, name, obj)` is thus equivalent to a call to
- * `bind(xc, name, obj, SERVICE)`.

c) The following text shall be considered to be added to the method `rebind(javax.tv.xlet.XletContext, String, Remote)`:

- * <p>The object shall be made visible to other applications in the same service context. A call
- * to `rebind(xc, name, obj)` is thus equivalent to a call to
- * `rebind(xc, name, obj, SERVICE)`.
- * Narrowing the scope of the binding (e.g. from GLOBAL to SERVICE) shall have the
- * same effect as a call to `unbind` for any applications which had references to
- * that object and which were in scope but which are now out of scope.

11 Security

11.1 Permission Request File Extensions

Clause 10 of TS 102 817 [6] shall be supported.

MHP-PVR terminals implementing TS 102 812 [2] shall support a new DTD for the permission request file as follows:

- The public identifier (referred to in TS 102 812 [2] as "PublicLiteral") to be used for specifying this DTD in document type declarations of the XML files is:
"-//DVB//DTD Permission Request File 1.1+PVR//EN"
- The URL for the SystemLiteral is:
"<http://www.dvb.org/mhp/dtd/permissionrequestfile-1-1-pvr.dtd>"
- The element "permissionrequestfile" shall be modified as below:
<!ELEMENT permissionrequestfile (file?,capermission?,applifecyclecontrol?,returnchannel?,tuning?, servicesel?,userpreferences?,network?, dripfeed?, persistentfilecredential*, applicationstorage?,smartcardaccess?, providerpermission?, recordingpermission?)>
Where the recordingpermission element is defined in clause 10 of TS 102 817 [6] and all other elements are unmodified.

MHP-PVR terminals implementing ES 201 812 [1] and TS 102 812 [2] shall support a new DTD for the permission request file as follows:

- The public identifier (referred to in ES 201 812 [1] as "PublicLiteral") to be used for specifying this DTD in document type declarations of the XML files is:
"-//DVB//DTD Permission Request File 1.0+PVR//EN".
- The URL for the SystemLiteral is:
"<http://www.dvb.org/mhp/dtd/permissionrequestfile-1-0-pvr.dtd>".
- The element "permissionrequestfile" shall be modified as below;
<!ELEMENT permissionrequestfile (file?,capermission?,applifecyclecontrol?,returnchannel?,tuning?, servicesel?,userpreferences?,network?, dripfeed?, persistentfilecredential*,recordingpermission?)>
Where the recordingpermission element is defined in clause 10 of TS 102 817 [6] and all other elements are unmodified.

12 System integration

12.1 TV-Anytime content referencing

NOTE: The present document does not define functional profiles or levels of the TV-Anytime specifications. One place where such activity is happening in the DTG TV-Anytime Test Bed project, see Bibliography.

12.1.1 Broadcast channel usage

Clauses 5, 6 and 7 of TS 102 323 [4] shall be supported.

12.1.2 Resolution by interaction channel

For resolution of content references using an interaction channel, clause 12.3 "Location Resolution Over Bi-directional Networks" of TS 102 822-6 [3] shall be supported.

13 Detailed platform profile definitions

Table 8: Detailed platform profile definitions

Area	Specification	Enhanced Broadcast Profile	Interactive Broadcast Profile and Internet Access Profile
Recording and playback process	6.1 Managing scheduled recording	M	M
	6.2 The recording process	M	M
	6.3 Managing completed recordings	M	M
	6.4 Playback	M	M
	6.5 Timeshift	M	M
	6.6 TV-Anytime	M	M
Metadata	7.1 TV-Anytime excluding 7.1.3 Transport by interaction channel	M	M
	7.1.3 Transport by interaction channel	-	M
Application model	8.1 Playback of recorded applications	M	M
	8.2 Service contexts and support for virtual channels	M	M
	8.3 Resource Management	M	M
	8.4 Modifications to MHP 1.0 application model specification	M	M
Application signalling	9.1.1 Signalling	M	M
	9.2 Signalling for application recording	M	M
	9.3 Extensions to application signalling	M	M
	9.4 Modifications to MHP 1.0 application signalling specification	M	M
DVB-J platform	10.1 PDR	M	M
	10.2 TV-Anytime	M	M
	10.3 Integration between PDR and TV-Anytime	M	M
	10.4 Version properties	M	M
	10.5 Extended semantics for MHP DVB-J Platform	M	M
	10.6 Modifications to MHP 1.0 DVB-J Platform	M	M
Security	11.1 Permission Request File Extensions	M	M
System integration	12.1 TV-Anytime content referencing excluding 12.1.2 Resolution by interaction channel	M	M
	12.1.2 Resolution by interaction channel	-	M
Registry of constants		M	M
Minimum Platform Capabilities	15 Minimum Platform Capabilities	M	M

Key	
-	Not required/Not applicable
O	Optional feature in the receiver
M	Mandatory feature in the receiver

14 Registry of constants

14.1 System constants

The following table defines the values for the system properties identifying the version of the present document supported by an implementation.

Field	Value
Major	1
Minor	0
Micro	2

15 Minimum Platform Capabilities

Clause 11 of TS 102 817 [6] shall be supported. Additionally MHP-PVR terminals shall support:

- For scheduled recording, the simultaneous recording of at least one video elementary stream, at least two audio elementary streams and at least two subtitle streams.
- The monitoring for changes in metadata accessed through the DVBDATABASE class of one query where all the results for that query are derived from modules signalled in the same DII. If results span more than one DII, at least one will be monitored. If multiple queries all map to data derived from the same DII then monitoring of all of them shall be supported.
- If the time-shift buffer has a fixed size, that fixed size shall be at least 5 minutes. If the time-shift buffer has a variable size, the MHP terminal shall have a means to clear at least 5 minutes which can be usable in the conformance testing process.

NOTE 1: This fixed size is defined for the purposes of conformance testing and should not be used by application or MHP terminal developers as being indicative of the capabilities of commercial products.

NOTE 2: The present document does not define minimum values for key parameters within the TV-Anytime specifications. Such activity is happening in the DTG TV-Anytime Test Bed project, see Bibliography.

Annex A (informative): Responsibilities of GEM Recording Specifications

A.1 Required responsibilities

The following table identifies where in the present document the required responsibilities listed in TS 102 817 [6] can be found.

Responsibility	Location of Definition
Which types of stream are to be considered as "recordable streams".	Clause 6.2.1 Identifying streams to be recorded
Minimum capabilities for the number of steams (or number of streams of each type) that a GEM recording terminal must be able to record.	Clause 15 Minimum Platform Capabilities
The definition of which applications are recordable in both scheduled and timeshift recording (which need not be the same).	Clause 6.2.2 Identifying and recording MHP applications for scheduled recording. Clause 6.5 Timeshift for timeshift recording.
The requirements on a GEM recording terminal to monitor for dynamic data and behaviour of applications during scheduled and timeshift recording (which need not be the same).	Clause 6.2.2 Identifying and recording MHP applications for scheduled recording. Clause 6.5 Timeshift for timeshift recording.
Requirements on reconstructing the dynamic behaviour of recorded applications during playback of scheduled and timeshift recordings (which need not be the same).	Clause 6.2.2 Identifying and recording MHP applications for scheduled recording. Clause 6.5 Timeshift for timeshift recording.
The definitions of which streams are to be recorded in scheduled and timeshift recording.	Clause 6.2.1 Identifying streams to be recorded for scheduled recording. Clause 6.5 Timeshift for timeshift recording.
How accurately the expiration period should be enforced by implementations.	Clause 6.3 Managing completed recordings
The definition of at least one protocol for transmitted time lines.	Clause 6.2 The recording process
The conditions when a JMF player or service context has a time-shift buffer attached.	Clause 6.5 Timeshift
A mechanism to associate security attributes with individual recording requests.	Clause 9.1.1 Signalling and clause 10.1.1 Common Core
The mechanism for resolving parent recording requests including setting the initial state of a parent recording request.	Clause 6.6 TV-Anytime
The events generated during playback when the start and end of a recording a reached.	Clause 6.4 Playback

A.2 Optional responsibilities

The following table identifies where in the present document the optional responsibilities listed in TS 102 817 [6] can be found or if they are not defined.

Responsibility	Definition
Mechanisms for controlling the extent to which one application can read or modify scheduled recordings and completed recordings made by another application.	Clause 10.1.1 Common Core
Sub-classes of RecordingListFilter to filter the list of recordings in ways not supported by the present document.	See org.dvb.pvr.navigation and org.dvb.tvanytime.pvr.navigation.
Rules governing which recordings an application can access.	Clause 10.1.1 Common Core
Additional JMF controls to be supported for RecordedServices or the contents of a timeshift buffer. Different sets of JMF controls may be specified for these two cases.	None defined in the present document.
Delays in re-starting applications after the return to normal play if this is believed to improve the end-user experience, for example when repeated cycles of fast-forward/play/fast-forward/play.	Clause 8.1 Playback of recorded applications
a mechanism to permit highly trusted applications to obtain instances of RecordingPermission whose action parameter is ""	No such mechanism defined in the present document.
that the optional behaviour defined in the class description of ServiceContextRecordingSpec, where the contents of the time-shift buffer are stored when the startTime parameter is in the past, becomes mandatory in that particular GEM recording specification.	Not mandatory in the present document.
Mechanisms for automatically removing requests from the list of recordings in a pending state if it appears the recording will never happen.	The validity period as and the requirements to respect it found in clause 6.1 Managing scheduled recording.
Mechanisms for automatically removing requests from the list of recordings in a failed state based on some criteria they define.	The validity period as and the requirements to respect it found in clause 6.1 Managing scheduled recording.
Any requirements to re-resolve ParentRecordingRequests after the request has first been made and to update the state accordingly	Clause 6.6 TV-Anytime

Annex B (informative): Bibliography

- The DTG TV-Anytime Test Bed project, see <http://www.dtg.org.uk/dtg/tva.html>

History

Document history		
V1.1.1	September 2007	Publication