ETSI TS 102 822-2 V1.2.1 (2004-09)

Technical Specification

Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("*TV-Anytime* Phase 1"); Part 2: System description

Reference
RTS/JTC-TVA-PH1-02-V2

Keywords
broadcasting, content, system, 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

Contents

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).

The present document is part 2 of a multi-part deliverable covering Broadcast and On-line Services: Search, select and rightful use of content on personal storage systems ("*TV-Anytime* Phase 1"), as identified below:

Part 1: "Phase 1 Benchmark Features";

Part 2: "System description";

Part 3: "Metadata";

Part 4: "Content referencing";

Part 5: "Rights management";

Part 6: "Delivery of metadata over a bi-directional network";

Part 7: "Bi-directional metadata delivery protection".

Introduction

The present document is based on a submission by the *TV-Anytime* forum (http://www.tv-anytime.org).

"TV-Anytime Phase 1" (TVA-1) is the first full and synchronized set of specifications established by the *TV-Anytime* Forum. TVA-1 features enable the search, selection, acquisition and rightful use of content on local and/or remote personal storage systems from both broadcast and online services.

The features are supported and enabled by the specifications for Metadata, Content Referencing, and Bi-directional Metadata Delivery Protection, TS 102 822-3 sub-parts 1 [1] and 2 [2], TS 102 822-4 [3], TS 102 822-6 sub-parts 1 [4] and 2 [5] and TS 102 822-7 [6] respectively. All Phase 1 Features listed in TV035r6 are enabled by the normative *TV-Anytime* tools specifications. This list of Phase 1 Features is to be used as guidance to manufacturers, service providers and content providers regarding the implementation of the Phase 1 *TV-Anytime* specifications.

There will be further *TV-Anytime* phases published and Business Models for Post-Phase 1 are currently being defined to include Private and public domains, portable recordable media, super distribution (legal sharing of content between consumers), peripheral device support and mobile devices, amongst others.

1 Scope

The present document shows the system behaviour of a *TV-Anytime* broadcast system with an interaction channel used for consumer response. It focuses on the use of the *TV-Anytime* content reference specification in combination with the *TV-Anytime* metadata specification in a system context. The present document will show examples of how to use both specifications both from static and dynamic viewpoints, i.e. it will highlight the parties involved in the processes and show the interaction between them. The present document will **not** show the use of rights management in the system: the *TV-Anytime* Rights Management specification will be included as soon as it is finalized.

To understand the present document, it is necessary for the reader to understand "*TV-Anytime* Requirements document R-2: the System Description". Since the present document applies TS 102 822-4 [3] and TS 102 822-3 sub-parts 1 [1] and 2 [2] are recommended reading. Both specifications enable the features in table 1 that will be highlighted in a system context in the present document.

The main part of the present document (excluding annex B) is that of a cookbook or white paper to the TS 102 822-4 [3] and TS 102 822-3 sub-parts 1 [1] and 2 [2]. It is an informative specification and has therefore not the intention to mandate certain system implementation solutions. Preferred solutions from a technology standpoint will be indicated to allow implementers to build efficient systems. Normative requirements that follow from writing the present document are inserted in the previously mentioned specifications. However, the *TV-Anytime* system needs certain capabilities from the underlying delivery system. Annex B, the only mandatory part of the present document, lists the requirements on the delivery system.

The present document is the second part of a series of gradually more complete system descriptions, enabling developers to make the most of *TV-Anytime* tools. Future versions of the present document will show more complete *TV-Anytime* system behaviour like the use of rights management and protection tools.

Table 1: Enabled feature set by TS 102 822-4 [3] and TS 102 822-3-1 [1]

Model 1 - Broadcast Model	**Support**
Use of ECG to find and capture broadcast content	Full
Search and selection of on-demand content with associated pricing information	Full
Capture and playback of audio, video and data (AVD)	Full
Cross linking of A/V content to related content	Part (see note 1)
Support of consumer preferences	Full
Content can be updated/replaced by newer in-coming versions	Part (see note 2)
Support for a variety of broadcast content types	Part (see note 3)
Support for all broadcast delivery mechanisms	Full
Multi-user preference support	Full
Model 2 - Consumer Response Model	**Support**
Updated listing/capture data can be delivered to "broadcast" analogue personal recorders (via return path or other mechanism)	Full
Updated listings/capture data can be delivered to "broadcast" PDRs	Full
Verification of usage of content on PDR	Part (see note 4)
Ability to collect usage data	Full
NOTE 1: Various types of content can be cross-linked using MediaLocator (see TS 102 822-3-1 [1]).The programme metadata does not contain a CRID for cross-linking to other programmes. NOTE 2: Entire programmes can be overwritten, but segments of programmes cannot be overwritten. NOTE 3: See TS 102 822-3-1 [1] for a list of supported content types. NOTE 4: Access to usage data is not specified by the current tools.	

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.

[1] ETSI TS 102 822-3-1: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime Phase 1"); Part 3: Metadata; Sub-part 1: Metadata schemas".

[2] ETSI TS 102 822-3-2: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime Phase 1"); Part 3: Metadata; Sub-part 2: System aspects in a uni-directional environment".

[3] ETSI TS 102 822-4: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime Phase 1"); Part 4: Content Referencing".

[4] ETSI TS 102 822-6-1: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime Phase 1"); Part 6: Delivery of metadata over a bi-directional network; Sub-part 1: Service and transport".

[5] ETSI TS 102 822-6-2: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime Phase 1"); Part 6: Delivery of metadata over a bi-directional network; Sub-part 2: Service discovery".

[6] ETSI TS 102 822-7: "Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime Phase 1"); Part 7: Bi-directional metadata delivery protection".

[7] ISO/IEC 15938-1 (2002): Information technology - Multimedia content description interface - Part 1: Systems.

[8] ISO 8601: "Data elements and interchange formats - Information interchange - Representation of dates and times".

[9] IETF RFC 1591: "Domain Name System Structure and Delegation".

[10] ISO 15706: "Information and documentation - International Standard Audiovisual Number (ISAN)".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

acquisition: retrieval of content

attractor: metadata element that is accessible by the consumer in order to aid in the content selection process, thus attracting the consumer

NOTE: Examples include the title and name of an actor in a television programme.

bi-directional: system that allows a two-way flow of content and/or information

data carousel: method for transmitting data over a broadcast channel in which data is cyclically transmitted

capture: storing the acquired content (to local storage)

consumer profile: data that represents the interests and preferences of the consumer

content: anything the viewer would like to consume (e.g. movies, games, TV programmes, radio programmes etc.)

content creator: producers of the content

content provider: entity that acts as the agent for and is the prime exploiter of the content

content reference: pointer to a specific content item

control flow: system related data e.g. consumer queries, transactional information, device capabilities, profile information etc.

functional unit: basic logical element, implementing a defined function of a *TV-Anytime* system

location resolution: process of establishing the address (location and time) of a specific content instance from its CRID

metadata: generally, data about content, such as the title, genre, summary of a television programme consumer preferences and viewing history data

V-ISAN: ISAN-based version identifier for audio-visual works

3.2 Abbreviations

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

CC	Content Creator
CRID	Content Reference IDentifier
NOTE:	An identifier for content that is independent of its location specified by TS 102 822-4 [3].
CSP	Content Service Provider
EPG	Electronic Programme Guide
NOTE:	A means of presenting available content to the consumer, allowing selection of desired content.
IPR	Intellectual Property Rights
ISAN	International Standard Audiovisual Number
ISO	International Organization for Standardization
LR	Location Resolution
PDC	Programme Delivery Control
PDR	Personal Digital Recorder
RAR	Resolving Authority Record
SMPTE	Society of Motion Picture and Television Engineers
SM	local Storage Management
SN	Search and Navigation
UI	User Interaction
V-ISAN	Versioned-International Standard Audiovisual Number

4 *TV-Anytime* system architecture

A simple *TV-Anytime* broadcast system can be viewed as containing three major elements: a service provider delivering the *TV-Anytime* service, a transport provider that carries the service and a piece of equipment in the home that stores the content and plays it back at the consumer's request. The "*TV-Anytime* R-2: System Description" document examines the mechanisms behind this simple model and gives a comprehensive functional reference model. This model adapted for the pure broadcast situation is depicted in figure 1. In this figure, a clustering of functions is indicated that is especially relevant in the broadcast case: it shows the "PDR" (Personal Digital Recorder).

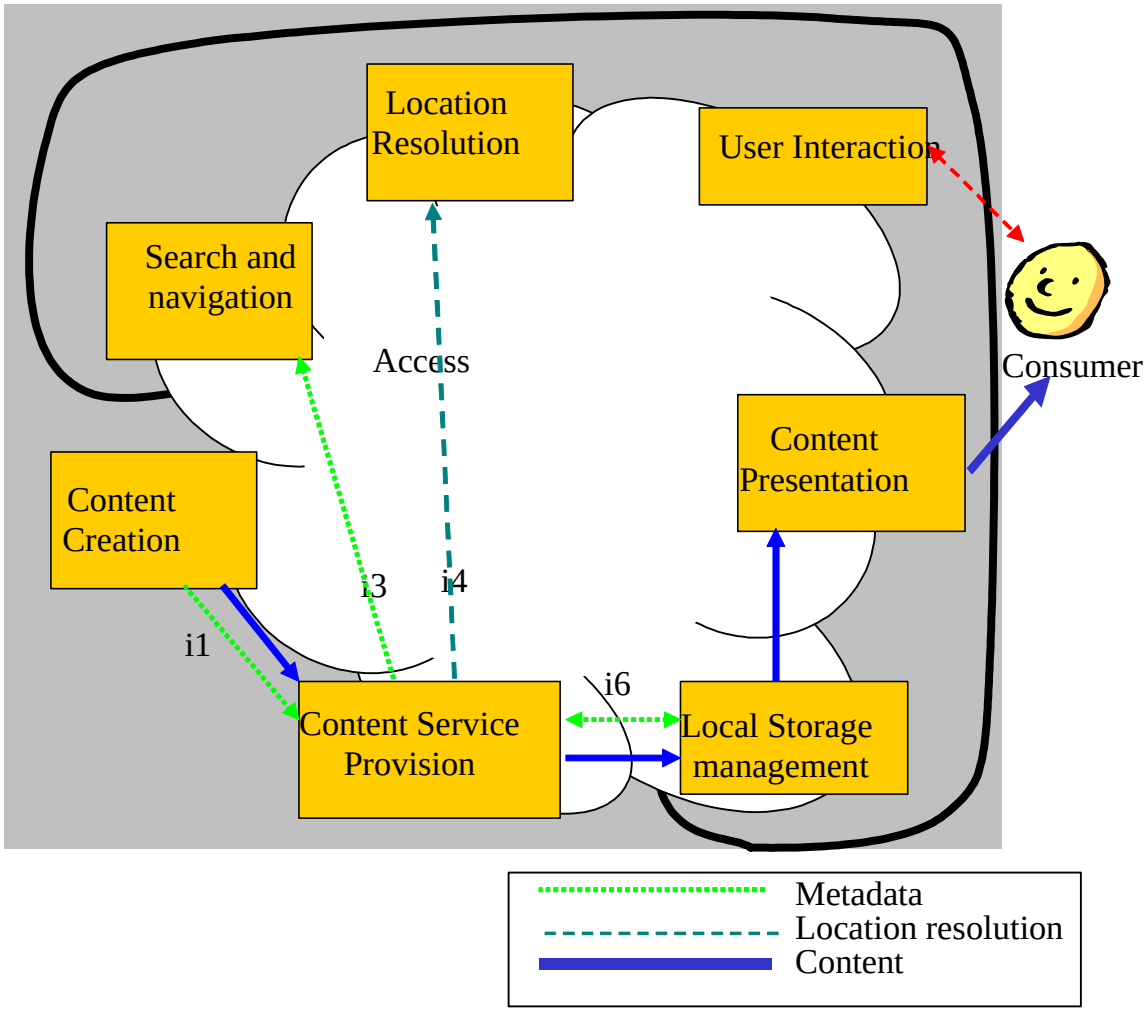

Figure 1: Broadcast model without rights management protection

Each of the boxes in the model is a function of the *TV-Anytime* system, and can be implemented in several different ways by several different service providers. Different physical implementations of the system will have different ordering of functionality in different physical devices (possibly in different locations). The present document gives a detailed description of possible system configurations. The arrows in the figure indicate information flows between functions, a complete description of these flows can also be found in the present document.

The broadcast model with a narrowband bi-directional channel supports the feature set listed in table 1. It is a pure broadcast model as far as content and associated data is concerned. In this broadcast model only three system functions are external to the PDR: content creation, content service provision and access. The bi-directional green link between user and service provider can be used to get usage history data or preference data from a consenting user. A movie studio or entertainment company could fulfil the role of content creator. A broadcaster would typically handle the repackaging, addition of metadata and broadcasting of the content: the content service provision function. A cable or satellite operator typically provides the access. The remaining functions reside in the PDR. The PDR can be considered as a real device at the consumer's premises that allows him to store and view content. In figure 1 the PDR is the grey area encompassing functions search and navigation, location resolution, user interaction, content presentation and local storage management.

This system will allow the user to search, select, locate and acquire content that he likes. The search and selection, e.g. by an EPG, will be on the basis of broadcast metadata that advertises the available content. One or more parties can put this metadata in the broadcast: the broadcaster, the content creator or a third party. The third party is not modelled in figure 1. An extension of this model showing third party operation will be discussed in clause 4.1.

The search and navigation will result after user or automatic selection in a Content Reference ID (CRID). The resolution function in the PDR, using the previously obtained content reference ID, results in a physical location of the content (e.g. a particular channel and time). Location resolution data must have been broadcast to allow the PDR to actually perform this translation from reference ID to in this example physical channel and time. The interfaces on the PDR will be subject to the appropriate rights management and protection policies that will be defined in a later version of the *TV-Anytime* specification series.

The "full interactive model" is described in "*TV-Anytime* R-1: The *TV-Anytime* environment". In this situation (see figure 2), the *TV-Anytime* consumer has a bi-directional link to other system functions like content service provision or search and navigation. In this case the following system functions could be external to the PDR: search and navigation, location resolution, content provision, content creation and access. Access will typically be provided by a telecommunications operator, there may however also be a broadcast operator providing a broadcast access function. Content creation can be done by entertainment companies as in the previous example, web designers or other interactive content designers (or even consumers!). Content service provision can be done by several parties e.g. webcasters, broadcasters, portal providers etc. The search and navigation function could be provided by a "web-EPG company" or TV-portal company. Location resolution could be done by a similar party. This is a new role or party that arises from this scenario. The PDR contains Storage, Content Presentation and User Interaction functions in one device.

The Rights Management and Protection part visualized in the figure that is currently covered is specified in TS 102 822-7 [6] "bi-directional metadata delivery protection".

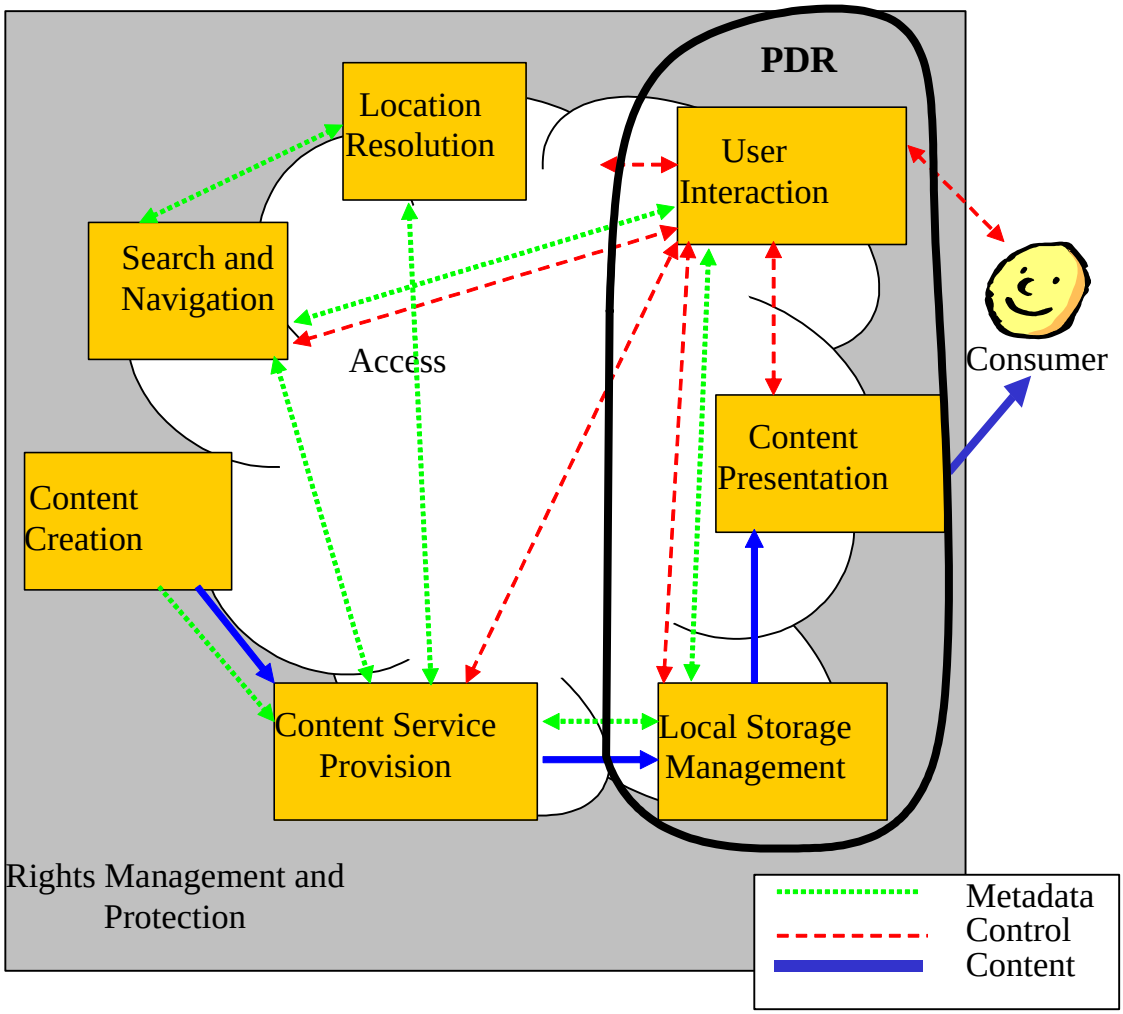

Figure 2: Full interactive model

4.1 Content referencing rationale

The purpose of content referencing is to allow acquisition of a specific instance of a specific item of content. For example, if a consumer sees an announcement on TV saying "there will be a new series on "Foxes in the cold" around Christmas", he may want to instruct his Personal Digital Recorder (PDR) to record the whole series. However the actual time and channel of airing of the episodes might be unknown to the PDR. In fact, the broadcaster may not know yet either. Still the viewer will want to make sure at this point that he does not miss the opportunity to acquire the content.

The ability to refer to content (in this example a series of programmes) independent of its location will provide this capability desired by the consumer. Whether that location is on a particular broadcast channel on some date and time, or on a file server connected to Internet, or wherever.

In this example, the PDR system would be provided with a reference for the series. In due time, the information required to link this reference to the individual episodes will be supplied to the PDR. Subsequently a specific date and time for each episode would be provided, so that the PDR would be able to acquire all of them.

This example demonstrates the purpose of content referencing - to provide the ability to refer to content independent of its location, and the ability to subsequently resolve such a reference into one or more locations where the content can be obtained.

4.2 Metadata rationale

Users or user-agents want to choose programmes to watch or record. To make that choice they need information like what is the title of this programme, what is it about, who are the actors, is it sci-fi? On the other hand, programme makers want to attract users to their content, by providing similar information. That is where metadata comes in: it is descriptive data about the content the user wants to consume. *TV-Anytime* content-related metadata is based on that assumption and is therefore largely "attractor" metadata, its goal being to provide choice to the user and means to service providers to advertise their content and services.

Clause 5 describes content referencing and the actors involved. Clause 6 describes the available metadata tools and their uses. Example walkthroughs and specific comments describing the dynamic system behaviour in the different phases of a *TV-Anytime* service lifecycle are described in clause 7 and clause 8, respectively.

5 *TV-Anytime* Content Referencing Scenarios

This clause introduces key concepts of content referencing, an extension of the static reference model introduced in clause 4 to model third party operation and possible scenarios of issuing and resolving references to items of content.

5.1 Content referencing key concepts

The key concept in content referencing is the separation of the reference to a content item - the CRID - from the information needed to actually retrieve the content item - the locator. The separation provided by the CRID enables a one-to-many mapping between content references and the locations of the deliverables. From a system perspective, content referencing and resolution lies between search and selection and actually acquiring the content. From the content referencing perspective, search and selection yields a CRID, which is resolved into either a number of CRIDs or a number of locators (the number may be one). A full discussion of content referencing is beyond the scope of the present document; rather it is the intention here to show how content referencing fits into the overall system. In the examples below, the syntax of a CRID and the syntax of a locator are employed. The syntax of a CRID is:

CRID://<authority>/<data>

Where <authority> takes the form:

<DNS name>

<DNS name> is a registered Internet domain name. The <DNS name> is case insensitive and must be a fully qualified name according to the rules given by RFC 1591 [9].

Some example authority names are:

www.broadcaster.com
ISP.net
www.commerce.com

The syntax of the locator is:

<transport mechanism>:<transport system specific>

The content referencing mechanism employs two key tables. The first is the Resolving Authority Record Table that maps the authority that issued the CRID to the Resolution Service Provider. The second table is the actual Resolution Table, which maps a CRID to another CRID or to a location. The resolution table may also contain information to link a locator to metadata describing that instance. Refer to TS 102 822-4 [3] *TV-Anytime* for a more detailed explanation of the concepts and tables involved.

5.2 Content referencing and unique content identification

Content referencing is the process of associating a token to a piece of content that represents its location where the content can be acquired. It is different from content identification, which creates an identifier that is the same regardless of its location.

A content reference is the token that is used by the PDR to acquire a piece of content once the user (or an agent working on their behalf) has selected it. The content reference is the "way pointer" from selection to acquisition.

A content identifier is an identifier that is created at the point just after the content is created with the idea that this identifier will always stay with the content. It allows metadata from multiple sources to all refer to the origination of the content. Whilst very useful, a content identifier is not designed for aiding acquisition of the content as it would require the (possibly globally centralized) body that created the content identifier to know about every instance of the content, and to be informed every time any of these locations change.

As there are already technologies being designed to fulfil the requirements of a content identifier, the *TV-Anytime* forum have chosen to design a content referencing token as this is an area that requires a global open standard.

The *TV-Anytime* specification uses the CRID as its token to represent the location of content. The CRID can be converted into either more CRIDs, or actual locations, by the process of location resolution.

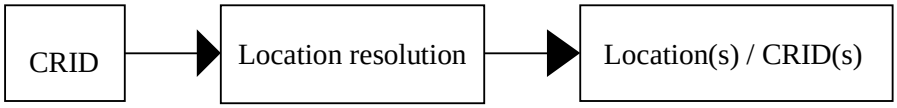

Figure 3: The location resolution process

The idea of a CRID being able to refer to other CRIDs is so that a CRID can represent a grouping of content (which is something that a content identifier cannot do). The group CRID can be used for representing any arbitrary group, an example of which is an entire series. A group CRID for an entire series would allow the PDR to acquire an entire series of programmes by just selecting one CRID to acquire.

One feature of the group CRID is that it means that many CRIDs may resolve to the same piece of content (as the content may be a member of many groups) which means that there might not be one unique CRID per content item.

The *TV-Anytime* defined CRID contains information about how to carry out the location resolution process. All CRIDs contain two parts, the first part is called an authority (which is the body that created this CRID) and the second part is data that has been created by the authority. A piece of information called the resolution authority record provides the mapping of resolution authority to the place where location resolution can take place.

An important feature of the *TV-Anytime* defined CRID is that it does not require a globally centralized body to assign CRIDs, as this was felt was impractical in that it may not scale well (e.g. in an Internet equipped PDR). From talking with various broadcasters it was also discovered that an advantage of a non-global registration system is that it allows material already in a broadcaster's catalogue to be broadcast without needing to register a globally unique identifier.

Another advantage of a de-centralized system is that the user can choose which authority that is closest to their personal tastes. For example one authority can choose that two programmes are the same, but another authority can specify that they are different. E.g. one is a widescreen presentation and the other is a pan and scan presentation of the same film, and the user can pick the authority that matches their personal views (e.g. they might consider widescreen and pan and scan versions identical).

5.3 CRID issuing Authority and resolution providers

The originator of a CRID is the party that actually creates the CRID and assigns it to reference an item of content: the Authority. The resolution provider is the party that provides the facility to resolve the CRID into a physical location in space and time. Three main actors can originate and resolve content reference identifiers: content creators, content service providers, and third parties. Third parties are not shown in figure 1, but can be modelled in quite easily. The next clause will describe the amended model, after that the possible combinations of actors are discussed.

5.3.1 Third party extension to static reference model

Figure 4 shows possible actors in the issuing and resolving of CRIDs. Two of those actors are already in the broadcast reference model shown in figure 1, the content creator and the content service provider. Third party operation is not explicitly modelled in figure 1. However, the model can be easily extended to cater for third party provisioning, both for CRID and resolution data, as well as metadata.

In figure 4 this extension is depicted. Only the existing interfaces are modelled in this figure. There may be a need for interfaces between the different functions in the content service provision function e.g. to enable the export of broadcasting schedules to the metadata provisioning function. Interfaces like that may be covered by future version of the *TV-Anytime* specification.

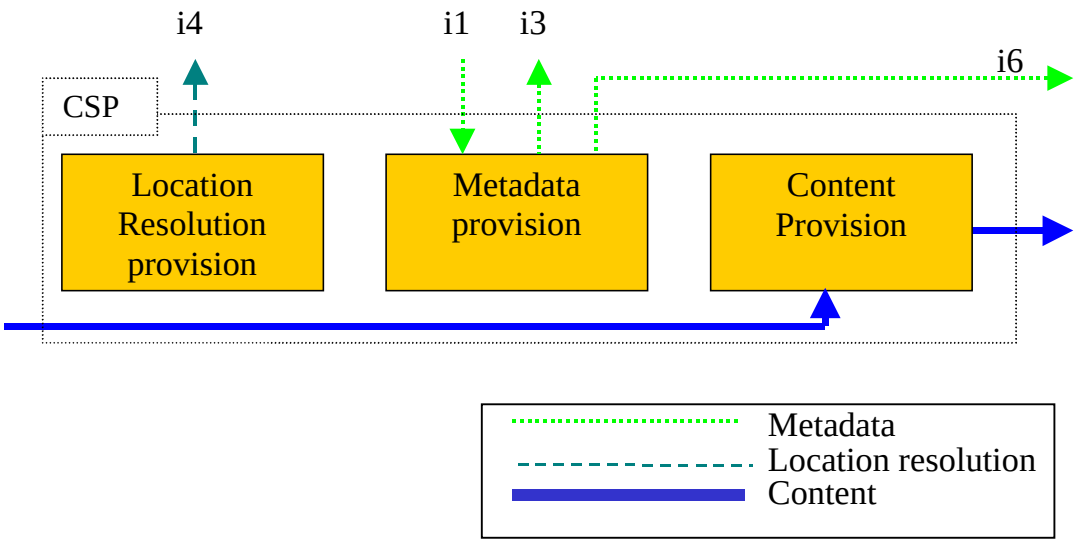

Figure 4: Extension to static reference model

The content service provisioning function in the overall model of figure 1 is split up in a number of different functions: a location resolution provision function, a metadata provision function, and a content provision function. For example in a service broadcast by broadcast XYZ, where XYZ is provisioning the repackaged content, there could be metadata from a third party with more extensive descriptions of XYZ content. This metadata could also be linked to CRIDs describing a different clustering of content: e.g. all episodes of a series with a certain actress in them. That same party could provide accompanying location resolution data for those CRIDs as well.

5.4 CRID issuing and resolving scenarios

In the most straightforward scenario, the originator of a CRID is also the resolution authority for that CRID. However, this relationship does not always hold. There are a number of scenarios where the CRID originator does not resolve the CRID itself. Table 2 depicts all possible CRID originators to CRID resolution authority scenarios. Table 2 which are likely and which are unlikely in the pure broadcast case. Unlikely in no way implies an impossible scenario.

Table 2: Originator of a CRID versus resolution of a CRID

	Content Creator resolves CRID	**Content Service Provider resolves CRID**	**Third Party resolves CRID**
Content Creator originates CRID	Likely	Likely	Likely
Content Service Provider originates CRID	Unlikely	Likely	Likely
Third Party originates CRID	Unlikely	Unlikely	Likely

The following clauses describe some of the scenarios in more detail.

5.4.1 CRID originated by content creator, resolved by content creator

In this simple scenario, the content creator creates the content and creates a CRID to reference that piece of content. The content creator also provides the resolving information to find that particular piece of content.

In the broadcast case, suppose the content creator is not the broadcaster, and the content in question is a drama entitled "Most Moving Drama Ever". The authority syntax might then be:

content.com

The CRID itself might take the form:

CRID://content.com/drama/MostMovingDramaEver

The string "drama/MostMovingDramaEver" is meaningful to the authority, i.e., the authority will be able to resolve this CRID when it is asked to do so. The content creator, having created the drama programme and having assigned it a CRID, needs to be able to broadcast the location resolution information to the PDR. This means it needs to have access to the broadcast channel and schedule information of the relevant broadcaster(s) involved. In the pure broadcast scenario, because there is no return channel, the location resolution takes place in the PDR itself. Finally, the content is broadcast and can be recorded on the local storage device and consumed later, or can be viewed at the time of broadcast.

When the broadcaster is also the content creator the scenario is simpler and described in clause 5.4.4.

5.4.2 CRID originated by content creator, resolved by content service provider

In this scenario, the content creator creates the content with an associated CRID. The content service provider is the resolution service provider.

Supposing the content creator is a motion picture studio, and the content in question is an action movie entitled "Best Action Movie Ever". The content service provider is a broadcaster. In this case the content service provider is acting as a proxy for the content creator. The content creator creates a CRID. It might look like this:

CRID://moviestudio.com/movies/BestActionMovieEver

The broadcaster, having purchased the movie from the studio for airing and having also acquired the CRID, broadcasts the location resolution information to the PDR. This information is contained in the "Resolution Tables" that map the CRID to the location. Also broadcast to the PDR are the Resolution Authority Records, one of which effectively includes a redirect, a record where the authority name and resolution service provider are different. In this example there is a Resolving Authority Record where the authority name is "moviestudio.com" and the resolution provider is "broadcaster.com".

In a uni-directional network, the location resolution takes place in the PDR. The consumer selects "Best Action Movie Ever" during some navigation or search process. The location resolution engine, having looked up the appropriate Resolving Authority Record, resolves the CRID whose authority is "moviestudio.com" by using the resolution service provider "broadcaster.com". The resolution service provided by "broadcaster.com" resolves the CRID to the actual location, the time and channel of the broadcast in the context of the service provider.

Finally, the movie is broadcast and can be recorded on the local storage device and consumed later, or can be viewed at the time of broadcast.

5.4.3 CRID originated by content creator, resolved by third party

In this scenario, the content creator creates the content and associated CRID. A third party resolves the CRID.

Supposing the content creator is a documentary production company, and the content in question is a documentary entitled "Incredible Documentary". Several broadcasters will carry this documentary over a period of time. In terms of location resolution the third party is acting as a proxy for the content creator. The production company creates a CRID. It might look like this:

CRID://documaker.com/IncredibleDocumentary

The third party might be an Electronic Content Guide service. The advantage of the third party in this case is that it can look across all service providers in the multiplex to resolve a CRID. The third party inserts the location resolution tables into the broadcast stream. Also inserted into the broadcast stream are the Resolving Authority Records, one of which contains the authority name "documaker.com" and the resolution service provider "res-service.ecg.com".

In a uni-directional network, the location resolution takes place in the PDR. The consumer selects "Incredible Documentary" during some navigation or search process. The location resolution engine searches the table of Resolving Authority Records and finds the one whose authority name matches the authority name in the CRID, in this example "documaker.com". The engine then uses the URL contained in the record to find the actual location resolution tables. In this way the CRID is resolved to a locator e.g.:

transport:channel5@8.00

Finally, the content is broadcast and can be recorded on the local storage device and consumed later, or can be viewed at the time of broadcast.

5.4.4 CRID originated by content service provider, resolved by content service provider

In this scenario, the content service provider acquires content and assigns its own CRID to reference that content. The content service provider is also the resolution service provider.

The content service provider could be a broadcaster, and the content in question is the movie "Best Action Movie Ever" from a motion picture studio. The motion picture studio, the content creator, may very well have its own CRID referencing the movie, but the content service provider decides not to use this. The broadcasters CRID might look like this:

CRID://broadcaster.com/movies/BestActionMovieEver

The broadcaster inserts the location resolution tables into the broadcast stream. Also inserted into the broadcast stream are the Resolving Authority Records, one of which contains the authority name "broadcaster.com" and the resolution service provider "broadcaster.com".

In a uni-directional network, the location resolution takes place in the PDR. The consumer selects "Best Action Movie Ever" during some navigation or search process. The location resolution engine searches the table of Resolving Authority Records and finds the one whose authority name matches the authority name in the CRID, in this example "broadcaster.com". The engine then uses the URL contained in the record to find the actual location resolution tables. In this way the CRID is resolved to a locator, e.g.:

transport:channel9@21.30

Finally, the content is broadcast and can be recorded on the local storage device and consumed later, or can be viewed at the time of broadcast.

5.4.5 CRID originated by content service provider, resolved by third party

In this scenario, the content service provider creates a CRID associated to content, but a third party is delegated to resolve the CRID. The content service provider could be a broadcaster. Suppose the content in question is the movie "Best Comedy Movie Ever". The motion picture studio, the content creator of the movie, may very well have its own CRID referencing the movie, but the content service provider decides not to use this. The broadcaster creates a CRID that might look like:

CRID://broadcaster.com/movies/BestComedyMovieEver

The third party might be a trusted agent such as an electronic content guide service provider. Either the broadcaster or the third party may create the location resolution tables as well as the Resolving Authority Records, one of which contains the name of the authority, "broadcaster.com", along with the name of the resolution service provider, "res-service.ecg.com". The Resolving Authority Records can be inserted into the broadcast. Thus, in terms of location resolution the third party is acting as a proxy for the content service provider, broadcaster.com. In a uni-directional network, the location resolution takes place in the PDR, while in a bi-directional network, the location resolution may take place on the server side. The consumer selects, as a result of some navigation or search process, "Best Comedy Movie Ever". The CRID resolution engine searches the table of Resolving Authority Records and finds the entry whose authority name matches the authority name in the CRID, "broadcaster.com" in this example, which is resolved in turn to give "res-service.ecg.com". The engine then uses this resolved information to find the actual location resolution tables, which are provided by "res-service.ecg.com". In this way, the CRID is resolved and the locator for the CRID is obtained. The content can now be captured.

5.4.6 CRID originated by third party, resolved by content service provider

In this scenario, a third party service creates e.g. a group CRID that references other CRIDs that in turn reference actual content. The third party could be an aggregator of some description. The content service provider agrees to be the resolution service provider for this third party because the third party service is particularly valuable.

Suppose the third party provides a CRID referencing all episodes of the "Star Journey" science fiction series. The CRID might look like this:

CRID://StarJourneyAggregator.com/AllEpisodesOfStarJourney

The broadcaster provides a Resolution Authority Record containing the authority name "StarJourneyAggregator.com" and the resolution provider "broadcaster.com".

The consumer, using some search and navigation process comes across the "All Episodes of Star Journey" item. The PDR looks up authority it finds in the CRID for this item. It finds that the resolution service provider is "broadcaster.com" and uses the URL to find the resolution tables. It resolves the CRID into a list of other CRIDs:

CRID://broadcaster.com/StarJourneyEpisode1

CRID://broadcaster.com/StarJourneyEpisode2

CRID://broadcaster.com/StarJourneyEpisode3

In this example, the broadcaster issued the returned CRIDs, however the third party could also have its own CRIDs for these episodes that a broadcaster knows how to resolve.

The various episodes are presented to the viewer for selection. The viewer selects "Star Journey Episode 2" and again the engine looks up the authority name in the Resolving Authority Record table. It finds that authority name "broadcaster.com" maps to resolution provider "broadcaster.com", and subsequently resolves the CRID to a list of alternate locations e.g.:

transport:channel9@21.30

transport:channel5@9.15

The most appropriate location is chosen depending on such factors as how soon the viewer wishes to watch the programme, recording conflicts if the programme is to be saved to local storage, the cost of one location versus the other etc.

Finally, the content is broadcast and can be recorded on the local storage device and consumed later, or can be viewed at the time of broadcast.

5.4.7 CRID originated by third party, CRID resolved by third party

In this scenario, a third party service creates e.g. a group CRID and references other CRIDs that in turn reference actual content. The third party is an aggregator of some description. The same or another third party is also the resolution service provider.

Suppose the third party provides a CRID referencing all nature documentaries on all channels within a multiplex. The CRID might look like this:

CRID://Aggregator.com/AllNatureDocumentaries

The third party provides a Resolution Authority Record containing the authority name "Aggregator.com" and the resolution provider "Aggregator.com". This is broadcast to the PDR along with the resolution tables, tables that the third party collates from schedule metadata it collects from all the content service providers in the multiplex.

The consumer comes across the "All Nature Documentaries" item in their Electronic Content Guide. The PDR looks up the authority it finds in the CRID for this item. It finds that the resolution service provider is "Aggregator.com" and uses the URL to find the resolution tables. It resolves the CRID into a list of other CRIDs:

CRID://Aggregator.com/FoxesInTheWild

CRID:// Aggregator.com/OceansOfTheWorld

CRID:// Aggregator.com/TheMapleTree

The various programmes are presented to the viewer for selection. The viewer selects "Oceans of the World" and again the engine looks up the authority name in the Resolving Authority Record table. It finds that authority name "Aggregator.com" maps to resolution provider "Aggregator.com", and subsequently resolves the CRID to a list of alternate locations e.g.:

transport:channel2@17.30

transport:channel7@21.00

The most appropriate location is chosen depending on such factors as how soon the viewer wishes to watch the programme, recording conflicts if the programme is to be saved to local storage, the cost of one location versus the other.

Finally, the content is broadcast and can be recorded on the local storage device and consumed later, or can be viewed at the time of broadcast.

5.5 Example of coding an ISAN or V-ISAN using a CRID

The ISO and SMPTE/ATSC have been actively working on the creation of a Versioned-International Standard Audiovisual Number (V-ISAN), which builds on ISO's original concept of an International Standard Audiovisual Number (ISAN). The goal of the new V-ISAN is to uniquely identify completed audio-visual works. In contrast with a CRID, the V-ISAN remains the same regardless of the provider of that content and would further allow comparisons between V-ISANs to determine that two pieces of content differ only by being a different version of the same root work or are different episodes of the same series.

The *TV-Anytime* Forum recognizes that some metadata and content providers have expressed interest in using the ISAN or the V-ISAN to identify the programmes they provide or reference in metadata. As such, the following CRID format is proposed to enable a CRID to be built using a known ISAN or V-ISAN.

An example CRID incorporating an ISAN will look like: **CRID://<authority>/isan<ISAN according to ISO 15706 [10]>**An example CRID containing a V-ISAN will look like:

CRID://<authority>/v-isan<V-ISAN>

In these examples the <authority> portion is as specified in TS 102 822-4 [3] and the <data> portion of the CRID is a (V-)ISAN, prefixed with the fixed string "isan" or "v-isan", respectively. In this case the normal use and semantics of the CRID are preserved. Namely, to convert this CRID into one or more location records the PDR contacts a location resolution service serving the Resolution Authority named in the <authority> portion of the CRID and passes the <data> portion, which in this case is a (V-)ISAN. However, because the data portion is clearly identified as a (V-)ISAN it also enables the PDR to make comparisons between CRIDs to determine if the referenced content is identical, different by version, or different by episode.

It is important to note that there is a significant difference between a CRID that is issued by a resolution authority and one that is constructed by the user interaction functionality in the PDR. There is an intention that a CRID that is issued will always be resolved by the relevant authority, whereas resolution of a constructed CRID is entirely speculative: one cannot rely on getting the location of the requested content.

Any unique ID for content can be used in a similar way as the (V-)ISAN in the above examples.

5.6 The relation between CRID and instance metadata

Instance description metadata is used to describe meaningful differences between specific instances of the same content i.e. instances of content that share the same CRID. For example, two instances of the same film where the instance description metadata indicates that one is in the original 16:9 aspect ratio and the other is in 4:3. Instance description metadata is linked to a particular event-related instance of content. For the full specification of instance description metadata see TS 102 822-3-1 [1] and TS 102 822-3-2 [2]. For a discussion of instance description metadata in the System context, see clause 6.3.3.

The *TV-Anytime* CRID is used to select and acquire an item of content independent of any particular location (time or place). In some cases however, the consumer may wish to acquire a location dependent version of a piece of content e.g. the 16:9 version of the film. To enable this functionality, the Content Referencing specification - TS 102 822-4 [3] - details an optional identifier called an Instance Metadata Identifier. This identifier is only guaranteed to be unique within the scope of the CRID to which it has been assigned. So it is permissible to assign the same identifier value to different CRIDs.

The PDR can use the instance metadata identifier to track changes in the scheduling of a specific instance of a piece of content. The following example illustrates the problem and the solution.

Suppose a PDR resolves a CRID:

crid://broadcaster.com/GreatMovie

to two locators:

dvb://123.5ac.3be;3e45@2001-12-07T12:00:00.00+01P02:10

dvb://487.2ee.3be;9e26@2001-12-09T12:00:00.00+01P02:10

Suppose further that the first locator is the 16:9 version of GreatMovie, and the second is the 4:3 version. The viewer decides that they want the 16:9 version, but it is not on for a couple of days yet so the PDR makes a note to acquire it. As it comes closer to the scheduled time (as indicated by the locator), the PDR again resolves the CRID to see if the time the film is going to be on has changed. This time the location resolution process yields two locators:

dvb://123.5ac.3be;3e45@2001-12-07T14:00:00.00+01P02:10

dvb://487.2ee.3be;9e26@2001-12-09T09:00:00.00+01P02:10

Both the 16:9 version and the 4:3 version have been rescheduled, and without instance metadata identifiers, the PDR would not be able to tell which is the location of the specific instance (the 16:9 instance) the viewer wants acquired.

With the use of instance metadata identifiers, each of the original two locators would be assigned an identifier. The instance metadata identifier appears in the location resolution tables and also appears in the corresponding instance metadata. So the first resolution of CRID:

crid://broadcaster.com/GreatMovie

yields:

dvb://123.5ac.3be;3e45@2001-12-07T12:00:00.00+01P02:10	imi:broadcaster.com/1
dvb://487.2ee.3be;9e26@2001-12-09T12:00:00.00+01P02:10	imi:broadcaster.com/2

The viewer selects the 16:9 version for acquisition, and this time the PDR takes note of the instance metadata identifier as well as the locator. As it gets closer to the scheduled time, the PDR once again resolves the CRID, and the location resolution process yields the following:

dvb://123.5ac.3be;3e45@2001-12-07T14:00:00.00+01P02:10	imi:broadcaster.com/1
dvb://487.2ee.3be;9e26@2001-12-09T09:00:00.00+01P02:10	imi:broadcaster.com/2

Once again the PDR is unable to tell which locator is the 16:9 instance of the film by just examining the locators, but this time it can tell that the first locator is the right one because the instance metadata identifier has not changed. It is the fact that the instance metadata identifier remains unchanged across schedule changes that solves this particular problem for the PDR.

The use of the imi might lead to unexpected effects. Consider the following. A consumer wishes to record the earliest showing of a certain programme. If he uses the imi to express that and the locator connected to that imi changes, it is possible that the PDR will not record the earliest showing.

As in the example above, suppose the user expresses that he wants to record the programme with imi:broadcaster.com/1, since this currently denotes the first showing:

dvb://123.5ac.3be;3e45@2001-12-07T19:00:00.00+01P02:10	imi:broadcaster.com/1
dvb://487.2ee.3be;9e26@2001-12-07T23:00:00.00+01P02:10	imi:broadcaster.com/2

If the programme gets rescheduled this might no longer be the case:

dvb://123.5ac.3be;3e45@2001-12-08T19:00:00.00+01P02:10	imi:broadcaster.com/1
dvb://487.2ee.3be;9e26@2001-12-07T23:00:00.00+01P02:10	imi:broadcaster.com/2

5.7 CRID Lifecycle

An authority who creates CRIDs and assigns them to pieces of content has the choice of keeping this assignment permanent, or making it temporary and re-assigning CRIDs to completely different pieces of content at a later date.

Assigning a CRID to a piece of content in a permanent manner has the advantage of allowing a PDR to always assume that every time it encounters the CRID it is referring to the same piece of content. However, this permanent assignment has the disadvantage that the CRID authority will need to keep records of this assignment to make sure it never re-uses the CRID for a different piece of content.

When CRIDs are re-used to refer to different pieces of content, there are a number of problems that may occur. The following scenarios demonstrate some of these problems:

1) A programme is being repeated a number of weeks apart. The first broadcast is selected for recording by a software agent whilst the user is on holiday. On returning from the holiday the user sees a trailer for the repeat (and does not necessarily know that it has already been recorded), which he selects to be recorded. The PDR obtains the programme CRID from the trailer, which it recognizes as the same CRID as one previously captured. The PDR now has two choices:

   a) Assume that the CRID refers to the same content concept as previously captured. There is therefore no need to record the programme and so the user should be informed that the programme is already available for viewing. If, in fact, the CRID has been reused to refer to a different programme concept in the intervening period, the PDR will fail to record the expected content.

   b) Assume that the trailer CRID has been reused since the previous one was captured and that the newly selected programme should be recorded. In this case, if the trailer CRID did still refer to the same content concept, the programme would be recorded twice and the PDR will have missed an opportunity to let the user watch the content sooner than it might otherwise have done so.

2) A programme has been recorded and the user chooses to watch it for the first time some months later. The cached metadata indicates that segmentation data is available (but the PDR did not originally acquire it). The user requests the segmented version, and to do this the PDR attempts to obtain the segmentation data from a *TV-Anytime* web service. If the CRID has been reused it is possible that segmentation data will be downloaded for the wrong content resulting in a confused user.

3) A content creator has issued a CRID and a third party web service offers enhanced metadata (such as programme reviews) on that content using the CRID provided by the content creator. If the content creator reuses the CRID it would cause programme reviews not to match the other metadata for the piece of content.

4) A PDR caches the metadata for a CRID along with the content. When the viewer comes to watch the content, the PDR sees that the content has parent CRIDs associated with it. The PDR would like to offer the user the chance to exploit this data - i.e. offer functionality like "record the whole series", "record the next programme". If the parent CRID has been re-assigned to a different programme grouping concept since it was originally issued, the PDR would acquire the incorrect series.

When CRIDs are assigned forever, there are also issues that need consideration. The following scenarios illustrate these issues:

1) If CRIDs are assigned forever the CRID author must maintain a history of all CRIDs issued, and knowledge of all metadata associated with the CRIDs. This may not be practicable for the following reasons:

   a) CRID authors working with broadcast schedules that extend over a short time interval may be unable to take into account all CRID allocations made during past scheduling/CRID authoring activities without incurring extra cost overheads that may be considered commercially unattractive.

   b) Allocation of CRID values may be made by resolution data authoring systems that will become obsolete or require re-setting from time to time.

   c) The CRID authority may decide to start issuing CRIDs for many items of content, such as trailers or adverts, to support independent acquisition of these short pieces of material. The nature of these short-lived publications may mean that CRID tracking by the author is not appropriate.

2) In many cases, several CRIDs may point to the same content. So, even if CRIDs are assigned forever, the content provider and user should still be encouraged to fully utilize the programme metadata as a means to describe the content, rather than just rely on the CRID.

3) Accidental re-use is likely to occur in a working system. Therefore a policy of reliance on non re-use may result in unpredictable or unknown behaviour, e.g. incorrect or missed acquisition of content.

As a result of the issues arising from the choice of CRID authoring policy, a working assumption for the unidirectional broadcast system would be to show caution when considering deliberately reusing CRIDs over short time intervals.

5.8 Harmonization of TS 102 822-4 and TS 102 822-6

TS 102 822-4 [3] specifies the requirements for unidirectional and bi-directional resolution of content references. It also provides an HTTP binding for bi-directional resolution over IP based networks. TS 102 822-6-1 [4] and TS 102 822-6-2 [5] meets all the requirements of TS 102 822-4 [3], and also provides a rich set of metadata queries. TS 102 822-6-1 [4] provides all the features of the HTTP binding in TS 102 822-4 [3] using a SOAP binding.

It is recommended that new server and client implementers using IP networks use the protocols defined in TS 102 822-6-1 [4] and TS 102 822-6-2 [5] in preference to TS 102 822-4 [3]. Note that this does not apply to the DNS based discovery of content referencing servers, as this is not superseded in TS 102 822-6-1 [4] and TS 102 822-6-2 [5]. TS 102 822-6-1 [4] and TS 102 822-6-2 [5] extends this DNS discovery mechanism to include metadata servers.

6 Metadata

6.1 Introduction

Metadata is generally defined as "data about data". Within the *TV-Anytime* environment, the most visible parts of metadata are the attractors/descriptors or hyperlinks used in electronic programme guides, or in Web pages. This is the information that the consumer or agent will use to decide whether or not to acquire a particular piece of content.

The *TV-Anytime* metadata system allows the consumer to find, navigate and manage content from a variety of internal and external sources including, for example, enhanced broadcast, interactive TV, Internet and local storage. It defines a standard way to describe consumer profiles including search preferences to facilitate automatic filtering and acquisition of content by agents on behalf of the consumer.

There is a need to associate metadata with content to facilitate human and automated searching for content of interest. Such metadata includes descriptive elements and attractors to aid the search process as well as elements essential to the acquisition, capture and presentation processes; content rights, formats, duration, etc. Many of these descriptive elements can be found in electronic programme guides and Web pages.

The process of creation and evolution of metadata for an individual content item may involve many organizations during the course of creation, distribution and delivery to the consumer. Thus, there is a clear need to define a common metadata framework and a standard set of metadata elements in order to ensure a high level of interoperability within the chain from content creation to content delivery.

6.2 XML - a common representation format

For the purpose of interoperability, the *TV-Anytime* Forum has adopted XML Schema as the common representation format for documentation of metadata. XML offers many advantages: it allows for extensibility, supports the separation of data from the application, and is widely used. In addition, powerful XML tools are now available such as XSL (XML Stylesheets), XQL (XML Query Language), and XML databases that can be used to process and manage XML data. As a textual format, XML tends to be rather verbose; however, a number of mechanisms are being developed to reduce the bandwidth when necessary. It is important to note that the XML representation of a *TV-Anytime* document is just that, *a representation*. It is *one possible representation of the metadata*; it is not the only representation of the metadata. There is no assumption that *TV-Anytime* metadata must be represented in XML format. Metadata could be represented by an optimized binary format to conserve bandwidth and aid rapid processing and mapping to a database. It is strongly recommended that if XML is used as exchange syntax for *TV-Anytime* metadata, then that XML should conform to the *TV-Anytime* Schema. This has obvious advantages in the business-2-business realm in addition to the business-2-consumer realm.

The following clauses introduce the *TV-Anytime* metadata schemas. They also provide snippets of XML instance documents. Basic knowledge of XML is needed in order to understand the following clauses.

6.3 The *TV-Anytime* metadata high level documents

All *TV-Anytime* metadata instance documents are grouped under a root element called "TVAMain".

6.3.1 Metadata structure

There are four basic kinds of metadata that a "TVAMain" element groups:

- Content description metadata.
- Instance description metadata.
- Consumer metadata.
- Segmentation metadata.

The diagram in figure 5 illustrates this relationship.

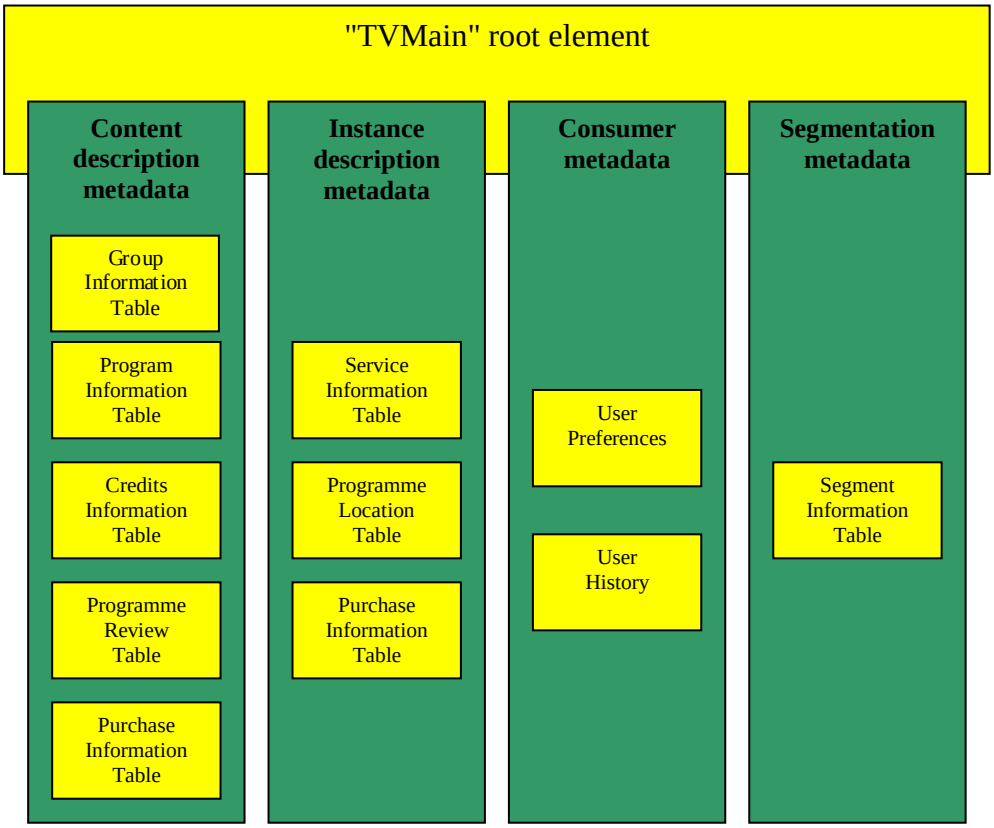

Figure 5: *TV-Anytime* documents with "TVA Main" as a root element

6.3.2 Content description metadata

Content Description metadata is divided into four areas:

- Descriptions of items of content e.g. television programmes. These descriptions are held in the ProgramInformationTable. They include things like the title of the programme, a synopsis, the genres it falls under and a list of keywords that can be used to match a search. The following example is a ProgramInformationTable containing a single ProgramInformation element. The example is not exhaustive.

```
<ProgramInformationTable>
   <ProgramInformation programId="crid://hbc.com/foxes/episode11">
      <BasicDescription>
         <Title type="main">
            The one where Fox jumps in the Potomac
         </Title>
         <Synopsis>
            Fox goes to Washington and jumps in the Potomac
         </Synopsis>
         <Keyword>Fox</Keyword>
         <Keyword>Washington</Keyword>
         <Keyword>Potomac</Keyword>
         <Genre href="urn:tva:metadata:cs:FormatCS:2004:3.5.7.3" type="main"/>
      </BasicDescription>
      <OtherIdentifier>guid://e41a-b456-a876-3e49</OtherIdentifier>
      <OtherIdentifier>urn:mpeg:mpeg21:diid:v-isan:29ef-94ba-53c4-3e7a-4ce8-e-5a45-98ec-
f</OtherIdentifier>
      <MemberOf crid = "crid://hbc.com/foxes/all" index = "11" xsi:type =
"EpisodeOfType"/>
   </ProgramInformation>
</ProgramInformationTable>
```

- Descriptions of groups of related items of content e.g. all episodes of "Foxes in the Wild". These descriptions are held in the GroupInformationTable. The following example is a GroupInformationTable containing a single ProgramInformation element. The example is not exhaustive.

```
<ProgramDescription>
   <GroupInformationTable>
      <GroupInformation groupId="crid://hbc.com/foxes/all">
         <GroupType xsi:type="ProgramGroupTypeType" value="series"/>
         <BasicDescription>
            <Title type="main">All episodes of Foxes ever</Title>
            <Synopsis>More Foxes than you can handle</Synopsis>
            <Keyword>Foxes</Keyword>
            <Keyword>all</Keyword>
            <Genre href="urn:tva:metadata:cs:FormatCS:2004:3.5.7" type="main"/>
         </BasicDescription>
            <MemberOf xsi:type="MemberOfType" crid="crid://hbc.com/comedy/all"/>
      </GroupInformation>
   </GroupInformationTable>
</ProgramDescription>
```

- A mapping of cast members to unique identifiers. The identifiers can be used in other metadata instances making searching easier. These descriptions are held in the CreditsInformationTable.

- Purchase information. This is held in the PurchaseInformationTable.

- Critical reviews of items of content. These descriptions are held in the ProgramReviewTable.

6.3.3 Instance description metadata

Instance Description metadata is divided into two areas:

- Descriptions of particular instances (locations) of content. These descriptions are held in the ProgramLocationTable. This metadata contains the scheduled time, but note that using this representation is *not* the preferred means of determining locations. The preferred means of determining locations is by resolving a CRID using the location resolution mechanism.

ProgramLocationTable contains records (elements) that are derived from ProgramLocationType (this is a base type, it is not instantiated directly, see TS 102 822-3-1 [1] and TS 102 822-3-2 [2]):

```
<ProgramDescription>
   <ProgramLocationTable>
      <BroadcastEvent serviceIDRef = "hbc100022311">
         <Program crid="crid://hbc.com/foxes/episode11"/>
         <ProgramURL>dvb://1.4ee2.3f5/</ProgramURL>
         <PublishedStartTime>2001-04-07T19:00:00.00+01:00</PublishedStartTime>
         <PublishedDuration>PT6H</PublishedDuration>
         <Live value="false"/>
         <Repeat value="true"/>
         <FirstShowing value="false"/>
         <LastShowing value="false"/>
         <Free value="false"/>
      </BroadcastEvent>
   </ProgramLocationTable>
</ProgramDescription>
```

It is possible to also include a BasicDescription element within BroadcastEvent. One use of this element is where an actor appearing in the programme has recently died, and the particular showing of the programme is a tribute. This extra information becomes an attractor for the programme. The synopsis of the programme is altered to reflect the fact that the programme features the deceased actor. It is more appropriate to change the synopsis for the instance, rather than the synopsis in the metadata attached to the CRID, as the "tribute" showing has a limited lifespan. Another use is where different instances have different technical attributes, such as aspect ratio or audio or video coding.

Optionally an imi can be used for each BroadcastEvent that can be used to link the instance metadata to the content referencing information.

Optionally, also here Purchase Information can be carried.

- Descriptions of services within a system. These descriptions are held in the ServiceInformationTable. Each description is encapsulated by a ServiceInformation element, illustrated in the example:

```
<ProgramDescription>
<ServiceInformationTable>
 <ServiceInformation serviceId="hbc100022311">
  <Name>HBC Channel 1</Name>
  <Owner>HBC</Owner>
 </ServiceInformation>
 <ServiceInformation serviceId="kgt1042062318">
  <Name>KGT Channel 9</Name>
  <Owner>KGT</Owner>
 </ServiceInformation>
</ServiceInformationTable>
</ProgramDescription>
```

6.3.4 Consumer metadata

Consumer metadata is divided into a number of areas:

- Details of a user's preferences or profile. This information is delivered by the UserPreferences description scheme, which provides rich representations of the particular types of content preferred or requested by the user. These descriptions are closely correlated with media descriptions, and thus enable users to efficiently search, filter, select and consume desired content. In the following example, the user ("Robert") prefers news programmes in English, when he is in Japan. The user also prefers comedy films reviewed and ranked by a particular film critic, as well as movies rated PG-13 by the MPAA (Motion Picture Association of America).

```
<UserDescription>
   <UserPreferences>
      <mpeg7:UserIdentifier protected="true">
         <mpeg7:Name xml:lang="en">Robert</mpeg7:Name>
      </mpeg7:UserIdentifier>
```

```
      <mpeg7:FilteringAndSearchPreferences>
         <mpeg7:ClassificationPreferences preferenceValue="10">
            <mpeg7:Language>en</mpeg7:Language>
            <mpeg7:Genre href="urn:tva:metadata:cs:FormatCS:2004:3.1.1"/>
         </mpeg7:ClassificationPreferences>
         <mpeg7:ClassificationPreferences preferenceValue="12">
            <mpeg7:Genre href="urn:tva:metadata:FormatCS:2004:3.5.7"/>
            <mpeg7:Review>
               <mpeg7:Rating>
                  <mpeg7:RatingValue>7</mpeg7:RatingValue>
                  <mpeg7:RatingScheme best="10" worst="1"
                     style="higherBetter"/>
               </mpeg7:Rating>
               <mpeg7:Reviewer xsi:type="mpeg7:PersonType">
                  <mpeg7:Name>
                     <mpeg7:FamilyName>Ebert</mpeg7:FamilyName>
                     <mpeg7:GivenName>Roger</mpeg7:GivenName>
                  </mpeg7:Name>
               </mpeg7:Reviewer>
            </mpeg7:Review>
            <mpeg7:ParentalGuidance>
               <mpeg7:ParentalRating
                  href="urn:mpeg:MPAAParentalRatingCS:PG-13">
                  <mpeg7:Name>PG-13</mpeg7:Name>
               </mpeg7:ParentalRating>
               <mpeg7:Region>us</mpeg7:Region>
            </mpeg7:ParentalGuidance>
         </mpeg7:ClassificationPreferences>
         <mpeg7:PreferenceCondition>
            <mpeg7:Place>
               <mpeg7:Name xml:lang="en">Tokyo</mpeg7:Name>
               <mpeg7:Region>jp</mpeg7:Region>
            </mpeg7:Place>
         </mpeg7:PreferenceCondition>
      </mpeg7:FilteringAndSearchPreferences>
   </UserPreferences>
</UserDescription>
```

- Details of a user's "click data", e.g. the actual usage history of a user's actions. UsageHistory description scheme provides a list of the actions carried out by the user over an observation period. This information can subsequently be used by automatic analysis methods to generate user preferences. An extensive example can be found in annex A.

6.4 Documents related through the CRID

Parts of a *TV-Anytime* document are related through the CRID. Metadata may be distributed across many *TV-Anytime* documents, but it is always possible to relate appropriate pieces through CRIDs.

6.4.1 Grouping

Programmes can belong to groups, and groups can belong to other groups. Linking programme descriptions with group descriptions using CRIDs reflects this relationship in the metadata, again.

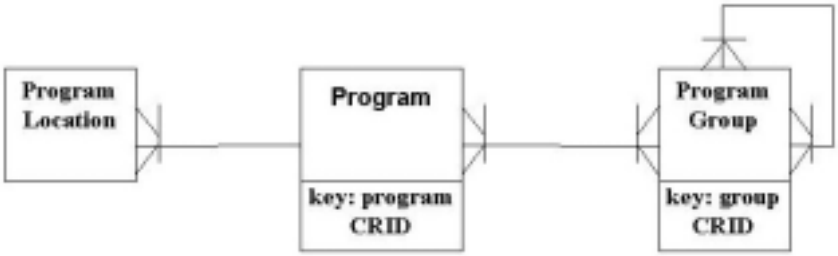

Figure 6: Programme descriptions related to group descriptions through the CRID

ProgramInformation elements are related to GroupInformation elements through the memberOf or episodeOf elements, e.g. the memberOf element contains a group CRID e.g. Foxes Episode 11 is a member of the Foxes group, which is a group that aggregates all episodes of Foxes. This supports the feature where a viewer can say "I like this. What is it? Are there more programmes like this?" By navigating up to the group the viewer may discover that the group is a member of another group and so forth. The higher one goes in the tree the more general the concepts become, e.g. moving from a specific episode of Foxes, to all episodes of Foxes, to all comedy shows, to all shows.

This upward pointing nature of group representation in the *TV-Anytime* metadata is the opposite of the content resolution process which is downward pointing (group CRIDs resolve into other CRIDs which resolve into locators).

6.5 *TV-Anytime* document structure

The following example illustrates the structure of a valid *TV-Anytime* document:

```
<TVAMain xmlns="urn:tva:metadata:2004"
   xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="urn:tva:metadata:2004 schemas/tva_metadata_v13_am1.xsd"
   version="03"
   xml:lang="en"
   publisher="..."
   publicationTime="2001-04-05T21:00:00.00+01:00">
   <CopyrightNotice>...</CopyrightNotice>
   <ProgramDescription>
   <ProgramInformationTable>...</ProgramInformationTable>
   <GroupInformationTable>...</GroupInformationTable>
   <ProgramLocationTable>...</ProgramLocationTable>
   <ServiceInformationTable>...</ServiceInformationTable>
   <CreditsInformationTable>...</CreditsInformationTable>
   <ProgramReviewTable>...</ProgramReviewTable>
   <PurchaseInformationTable>…</PurchaseInformationTable>
   </ProgramDescription>
   <UserDescription>
      <UserPreferences>...</UserPreferences>
      <UsageHistory>...</UsageHistory>
   </UserDescription>
</TVAMain>
```

Many of the elements are optional, so the following examples are also valid documents:

```
<TVAMain version="03" xml:lang="en" publisher=".." publicationTime="..">
 <CopyrightNotice>...</CopyrightNotice>
 <ProgramDescription>
  <ProgramInformationTable>...</ProgramInformationTable>
 </ProgramDescription>
</TVAMain>
```

```
<TVAMain version="03" xml:lang="en" publisher=".." publicationTime="..">
 <CopyrightNotice>...</CopyrightNotice>
 <ProgramDescription>
  <GroupInformationTable></GroupInformationTable>
 </ProgramDescription>
</TVAMain>
```

```
<TVAMain version="03" xml:lang="en" publisher=".." publicationTime="..">
 <CopyrightNotice>…</CopyrightNotice>
 <UserDescription>
  <UserPreferences>…</UserPreferences>
  <UsageHistory>…</UsageHistory>
 </UserDescription>
</TVAMain>
```

6.6 Mandatory and optional elements

The *TV-Anytime* XML Schema contains many elements that are optional and some that are mandatory. The diagram shows the mandatory parts of ProgramInformation:

```
<ProgramInformationTable>
 <ProgramInformation programId="crid://hbc.com/foxes/episode1">
  <BasicDescription>
   <Title type="main">
    The one where Fox jumps in the Potomac
   </Title>
   <Synopsis>
    Fox goes to Washington and jumps in the Potomac
   </Synopsis>
  </BasicDescription>
 </ProgramInformation>
</ProgramInformationTable>
```

6.7 Delivery of metadata over a unidirectional environment

The *TV-Anytime* forum has specified a generic solution for the carriage of metadata over unidirectional environments. Unidirectional environments are defined as being environments where content and metadata are delivered from the transmitting device (head-end) to the terminal device (PDR) over a one-way link and where no communication is possible from the PDR to the head-end. The *TV-Anytime* solution is not specific to a particular transport layer and has been designed so as to be used within any unidirectional delivery system. The requirements needed to be fulfilled by such a system in order to be able to implement this solution are listed in annex B.

The *TV-Anytime* solution has been designed to support the following methods of acquisition of the *TV-Anytime* metadata sent out over a unidirectional network:

- Method 1: Acquire from the metadata stream and cache the data to disk with the receiver provides its own methods of navigation.
- Method 2: Use the TVA Indexing solution to enable online navigation of the metadata stream.
- Method 3: Cache both TVA indexing information and data to disk to provide an enhanced version of method 2.

Accordingly the delivery of a *TV-Anytime* description, namely the actual document containing all the *TV-Anytime* metadata from a certain metadata provider to be sent out at a certain time, is viewed as made of five distinct processes. Figure 7 shows the processes associated with the delivery of metadata. Those specified by the *TV-Anytime* delivery solution are shown in grey.

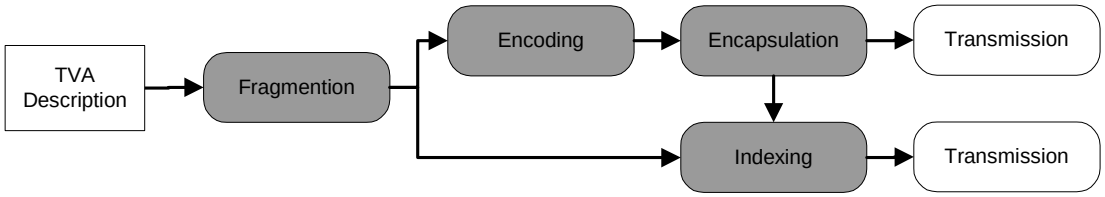

Figure 7: Processes associated with the delivery of metadata

Fragmentation is the generic decomposition mechanism of a TVA metadata description into self-consistent units of data, called *TVA fragments.* A fragment is the ultimate atomic part of a *TV-Anytime* metadata description that can be transmitted independently to a terminal. A fragment shall be self consistent in the sense that:

- It shall be capable of being updated independently from other fragments.
- The way it is decoded, processed and accessed shall be independent from the order in which it is transmitted relative to other fragments.
- The decoding of a fragment and its addition to the partial description shall give a *TV-Anytime* schema valid description. Note that a partial description must have at least the fragment delivering the root element (`TVAMain`).

A number of normative TVA fragment types have been defined as follows:

- *TVAMain* fragment which contains the root element of the description.
- *ProgramInformation* fragment containing metadata for a given content.
- GroupInformation fragment containing metadata for a given group of contents.
- *OnDemandProgram* and *OnDemandService* fragment for the description of on-demand instances of contents.
- *BroadcastEvent* fragment, *Schedule* fragment and *ServiceInformation* fragment used for the description of broadcast instances of contents and of the services where they are available.
- For the *CreditInformation* metadata, *PersonName* and *OrganizationName* fragments.
- *Review* fragment to contain review of a given content.
- *Purchase Information* fragment to contain price information.
- For the ClassificationSchemes metadata, *CSAlias* fragment and *ClassificationScheme* fragment.
- For the Segmentation metadata, *SegmentInformation* fragment and *SegmentGroupInformation* fragment.

Encoding is the process that enables the efficient (in terms of bandwidth, navigability and updating) delivery of data within a unidirectional environment. It consists in representing the TVA metadata fragments in a binary format. *TV-Anytime* has chosen the MPEG-7 BIM BiM method as defined in ISO/IEC 15938-1 [7] (MPEG-7 Systems part) as the preferred method that would facilitate wide interoperability. However *TV-Anytime* appreciates that in some controlled environments, it may be desirable to enable the delivery of metadata using alternate encoding systems. To allow this, appropriate hooks are provided where necessary and the means to indicate the method of encoding used.

Encapsulation is the process, which enable the grouping of a number of binarised fragments together in a "container" ready for transmission. It associates to these fragments further information enabling a receiving device to manage them. In particular it allocates to each fragment a unique identifier within the TVA metadata fragment stream and a version number so as to enable the monitoring of possible updates.

To conclude, indexing is the optional mechanism seen as suiting especially situations when TVA metadata is to be delivered to receivers that have limited processing and storage capabilities. As within a *TV-Anytime* metadata fragment stream there is are likely to be many hundreds of fragments, indexing provides a mechanism for locating information from within this stream. It allows multiple views on the data set of a TVA metadata description and enables a device to quickly find a fragment of interest. Indexing structures sent out along with the TVA metadata fragment stream provide direct access to each TVA fragment by listing the values of a particular node (the index's key fields) and describing where the matching fragment(s) can be found over the delivery layer. Multiple indices can point to the same fragment, each using a different node as a key field. The indexing structures when provided are transmitted using the generic container format defined by the encapsulation mechanism.

For the transmission, *TV-Anytime* does not define the way in which these containers should be carried, as this is specific to the delivery system. However in the specification of a Container, consideration has been given, to enable the container to be easily mapped on to standard delivery methods. For example in an MPEG-2 environment, the containers may be conveyed using Sections, objects within a DSM-CC U-U Object Carousel or modules within a DSM-CC Data Carousel.

6.8 Notes on Schema Extension

The *TV-Anytime* Schema defined in TS 102 822-3-1 [1], provides a standard way of describing common data structures required within a PDR environment. However there may be instances where third parties may wish to extend the TVA schema to provide enhancements to existing data types, or to introduce completely new data types, providing additional functionality.

This can be achieved in a backward compatible way using standard XML Schema methods. TS 102 822-3-2 [2] defines a subset of these XML Schema methods which are applicable within the context of a TVA Schema.

Note that the declaration of new data types must occur in a separate schema document and have their own unique namespace.

6.8.1 Polymorphism of existing type by Inheritance with extension

To extend an existing TVA data type one uses the XML Schema, derive by extension mechanism. So for example if a provider wish to add a new element called MyData to a standard TVA ProgramInformationType, he would define a new type as follows:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:tva="urn:tva:metadata:2004"
elementFormDefault="qualified" attributeFormDefault="unqualified">
   <xs:import namespace="urn:tva:metadata:2004"
schemaLocation="./tva_metadata_v13_am1.xsd"/>
<xs:complexType name="MyDataType">
   <xs:complexContent>
      <xs:extension base="tva:ProgramInformationType">
         <xs:sequence>
            <xs:element name="MyData" type="xs:string" />
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>
</xs:schema>
```

To use the new data type within an instance document, one makes use of the XML Schema "type" attribute to declare the actual data type as follows:

```
<tva:TVAMain xmlns:tva="urn:tva:metadata:2004"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ExampleTvaExtension.xsd">
   <tva:ProgramDescription>
      <tva:ProgramInformationTable>
         <tva:ProgramInformation programId="CRID://bbc.com/films/titanic"
xsi:type="MyDataType">
            <tva:BasicDescription>
               <tva:Title>Titanic</tva:Title>
```

```
            </tva:BasicDescription>
            <MyData>xxxxxxxxx</MyData>
         </tva:ProgramInformation>
      </tva:ProgramInformationTable>
   </tva:ProgramDescription>
</tva:TVAMain>
```

If no "type" is declared for the extended type the system assumes that it is of the base type, which in this case is ProgramInformationType.

It should be noted that all new data elements occur at the end of the extended data type. In the case where an extension adds new attributes, then these attributes can occur in any order within the extended element.

6.8.2 Polymorphism of existing type by Inheritance with restriction

To restrict an existing TVA data type one uses the XML Schema, derive by restriction mechanism. So for example if a provider wishes to create a new data type called MyDataType which removes all optional elements from the standard TVA ProgramInformationType, he would define a new type as follows:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:tva="urn:tva:metadata:2004"
elementFormDefault="qualified" attributeFormDefault="unqualified">
   <xs:import namespace="urn:tva:metadata:2004"
schemaLocation="./tva_metadata_v13.xsd_am1"/>
<xs:complexType name="MyDataType">
   <xs:complexContent>
      <xs:restriction base="tva:ProgramInformationType">
         <xs:sequence>
            <xs:element name="BasicDescription" type="tva:BasicContentDescriptionType"/>
         </xs:sequence>
         <xs:attribute name="programId" type="tva:CRIDType" use="required"/>
         <xs:attributeGroup ref="tva:fragmentIdentification"/>
      </xs:restriction>
   </xs:complexContent>
</xs:complexType>
</xs:schema>
```

To use the new data type within an instance document, one makes use of the xsi:type attribute to declare the actual data type as follows:

```
<tva:TVAMain xmlns:tva="urn:tva:metadata:2004"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ExampleTvaExtension.xsd">
   <tva:ProgramDescription>
      <tva:ProgramInformationTable>
         <tva:ProgramInformation programId="CRID://bbc.com/films/titanic"
xsi:type="MyDataType">
            <tva:BasicDescription>
               <tva:Title>Titanic</tva:Title>
            </tva:BasicDescription>
         </tva:ProgramInformation>
      </tva:ProgramInformationTable>
   </tva:ProgramDescription>
</tva:TVAMain>
```

If no "type" is declared the system assumes that it is of the base type, which in this case is ProgramInformationType.

7 Cookbook examples and scenarios

This clause describes the phases identified in a *TV-Anytime* System. It is followed by examples to give an overview of how a system might work. Further details and issues that arise from this example are identified in the next clause. The examples will cover usage of both content referencing and metadata (including the usage of metadata over a bi-directional link).

7.1 *TV-Anytime* dynamic phases

Phases in a *TV-Anytime* session are depicted in figure 8. A more detailed explanation of these phases is covered by "*TV-Anytime* R-2: System Description".

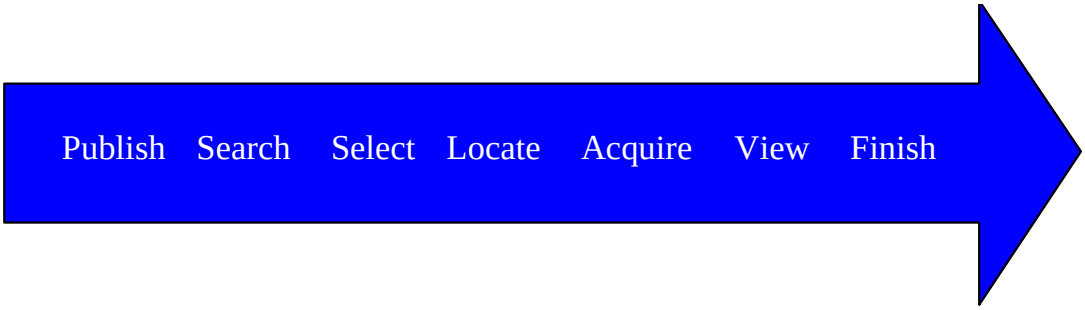

Figure 8: Phases in a *TV-Anytime* system

The next clause presents an example that shows, per phase, the steps that are to be taken in a *TV-Anytime* system.

7.2 Example: Record every episode of this programme series in the broadcast case

This example shows how the current *TV-Anytime* system may work using the TS 102 822-4 [3], TS 102 822-3-1 [1] and TS 102 822-3-2 [2]. The example is intended to give an overview of the system; more specific issues per phase will be covered later in the next clause.

Publish

A content service provider will publish a CRID that represents a programme series, and CRIDs that represent the constituents of that programme series. The same or different service provider will publish metadata that describes this series and its constituent episodes. The same or different service provider will publish location resolution data that describes where and when the constituent episodes of this series may be acquired. The series may be available from multiple content service providers.

In this example we will use a comedy show "Fox" which has two episodes. The included XML snippets show an almost minimal way to describe this show and its episodes. Three metadata tables are needed to describe the relations for the Fox show. The GroupInformation table that holds information for all episodes of Fox and two ProgramInformation tables that contain information for the different episodes.

The link between the group and the episodes is made by the content referencing system: if the Group CRID "//hbc/foxes/all" is put to the resolution engine in the PDR, it will come back with both programme CRIDs. The link between programmes and the group is being made by the <memberOf> element in the ProgramInformation table.

```
<ProgramDescription>
   <ProgramInformationTable>
      <ProgramInformation programId="crid://hbc.com/foxes/episode1">
         <BasicDescription>
            <Title type="main">
            The one where Fox jumps in the Potomac
            </Title>
            <Synopsis length="short">
```

```
            Fox goes to Washington and jumps in the Potomac
            </Synopsis>
         </BasicDescription>
         <MemberOf xsi:type="EpisodeOfType" crid="crid://hbc.com/foxes/all" />
      </ProgramInformation>
      <ProgramInformation programId="crid://hbc.com/foxes/episode2">
         <BasicDescription>
            <Title type="main">
            The one where Fox drowns in the Lake of Geneva
            </Title>
            <Synopsis length="short">
            Fox goes to Geneva and tries to climb the fountain
            </Synopsis>
         </BasicDescription>
         <MemberOf xsi:type="EpisodeOfType" crid="crid://hbc.com/foxes/all"/>
      </ProgramInformation>
   </ProgramInformationTable>
   <GroupInformationTable>
      <GroupInformation groupId="crid://hbc.com/foxes/all" ordered="true" numOfItems="2">
         <GroupType xsi:type="ProgramGroupTypeType" value="show"/>
         <BasicDescription>
            <Title type="main">
            All episodes of Foxes ever
            </Title>
            <Synopsis length="short">
            More Foxes than you can handle
            </Synopsis>
         </BasicDescription>
         <MemberOf xsi:type="MemberOfType" crid="crid://hbc.com/comedy/all"/>
      </GroupInformation>
   </GroupInformationTable>
</ProgramDescription>
```

To allow PDRs to build for, example, an EPG or to inform the user of the approximate schedule of an airing, InstanceMetadata can be used. This is useful, for example, when a user is interested in watching a programme in a non-time shifted manner. The BroadcastEvent table in the instance metadata is used for that purpose. It is *NOT* there to signal to the PDR where and when a particular programme really can be found: that is the job of content referencing and location resolution mechanism. As stated before, the metadata is used mainly for attraction purposes. An example XML snippet of a BroadcastEvent table is given below:

```
<ProgramLocationTable>
<BroadcastEvent serviceIDRef = "hbc100022311">
 <Program crid="crid://hbc.com/foxes/episode1"/>
<BroadcastURL>dvb://1.4ee2.3f4/</BroadcastURL>
 <EventDescription>
  <PublishedTime>2001-04-05T21:00:00.00+01:00</PublishedTime>
  <PublishedDuration>PT3H</PublishedDuration>
 </EventDescription>
 </BroadcastEvent>
</ProgramLocationTable>
```

Search

One example of a search is a user searching for the title of a series, e.g. "Foxes", that he is interested in. The result of the search is a list of matching titles and associated identifiers (CRIDs). To refine his search further, the user must examine other metadata that can be attached to the CRID. The user can refine his search further to identify the particular series he wants to acquire, e.g. "Foxes in the Wild". The search may then be refined even further, e.g. by specifying PPV or free-to-air or quality.

Another example is that a user likes the programme he is currently viewing and wants to see more programmes like this one. First the system must find the CRID of the current programme being viewed. If the programme is played from disk then the system should have stored the identity of the programme and associated metadata. If the programme is "live" then the system must be able to find the CRID of the programme on the current channel. Once the CRID has been found then the system must find the metadata associated with this CRID and interrogate it. In this example the programme is a member of a series. The user reads the description of the series and decides to record the whole series.

Other search mechanisms, based on the UserPreferences metadata are also possible. The search intention can for example be captured in the UserPreference DS.

In this example the user searches for "Fox". The PDR in this example examines the title and synopsis fields of the Group and Program Information table and outputs as a result of his search three descriptions, one of the group, and two of the episodes. Other implementations could also search other metadata elements like keywords or genre. The user selects episode 2 and reads the synopsis. For this example we assume that after viewing info about this episode the user wants to record the whole series.

Select

For our example we assume that the PDR will offer the user the option to record the whole series. At this point to get the whole series the PDR examines the MemberOf element in the ProgramInformation table and sees that the episode is part of a show called "Fox" with CRID "//hbc.com/foxes/all". With this CRID available it will try to locate the actual episodes in the next phase.

At this point the usage history metadata table in the PDR could be updated, showing that the user has made a selection. An example XML snippet is below:

```
<UserDescription>
   <UsageHistory id="usage-history-001" allowCollection="true">
      <mpeg7:UserIdentifier protected="true">
         <mpeg7:Name xml:lang="en">John Doe</mpeg7:Name>
      </mpeg7:UserIdentifier>
      <mpeg7:UserActionHistory id="useraction-history-001"
         protected="false">
         <mpeg7:ObservationPeriod>
            <mpeg7:TimePoint>2001-02-02T18:00-08:00</mpeg7:TimePoint>
            <mpeg7:Duration>PT96H</mpeg7:Duration>
         </mpeg7:ObservationPeriod>
         <mpeg7:ObservationPeriod>
            <mpeg7:TimePoint>2001-02-02T18:00-08:00</mpeg7:TimePoint>
            <mpeg7:Duration>PT6H</mpeg7:Duration>
         </mpeg7:ObservationPeriod>
         <mpeg7:UserActionList id="ua-list-001" numOfInstances="1"
totalDuration="PT2H30M">
            <mpeg7:ActionType href="urn:tva:metadata:cs:ActionTypeCS:2002">
               <mpeg7:Name>Record</mpeg7:Name>
            </mpeg7:ActionType>
            <mpeg7:UserAction>
               <mpeg7:ActionTime>
                  <mpeg7:MediaTime>
                     <mpeg7:MediaTimePoint>2001-02-02T19:00:00</mpeg7:MediaTimePoint>
                     <mpeg7:MediaDuration>PT1H</mpeg7:MediaDuration>
                  </mpeg7:MediaTime>
               </mpeg7:ActionTime>
               <mpeg7:ProgramIdentifier organization="TVAF"
type="CRID">crid://hbc.com/foxes/all</mpeg7:ProgramIdentifier>
            </mpeg7:UserAction>
         </mpeg7:UserActionList>
      </mpeg7:UserActionHistory>
   </UsageHistory>
</UserDescription>
```

This usage history could also be used by the PDR to fill the user preference metadata tables. A more extensive example of usage history can be found in annex A.

As far as the user is concerned, the system will now autonomously make the content available at some point in the future.

Locate

Once the particular series has been chosen the series must be "resolved" to its constituent episodes. Given the CRID for the series the location resolution functional unit will return a list of CRIDs that refer to each episode. This relies on the fact that the location resolution data is made available to the box.

The resolution process continues until each of the episodes is then resolved to locations (channel/time/duration in the broadcast case). For each episode there may be several locations, e.g. repeats. These locations contain the same content as far as the service provider is concerned.

For our example, the show "Fox" has the following resolution tables associated with it:

```
<ContentReferencingTable>
 <!-- CRID resolution to other CRIDs -->
 <Result CRID="crid://hbc.com/foxes/all"
   status="resolved" complete="true" acquire="all">
  <CRIDResult>
   <Crid>crid://hbc.com/foxes/episode1</Crid>
   <Crid>crid://hbc.com/foxes/episode2</Crid>
  </CRIDResult>
 </Result>
 <!-- CRID resolution to locators -->
 <Result CRID="crid://hbc.com/foxes/episode1" status="resolved"
   complete="true" acquire="all">
  <LocationsResult>
   <Locator>dvb://1.4ee2.3f4;4f5@2001-04-05T21:00:00.00+01:00/PT00H45M
   </Locator>
  </LocationsResult>
 </Result>
 <Result CRID="crid://hbc.com/foxes/episode2"
   status="cannot yet resolve" complete="true"
   acquire="all" reresolveDate = "2001-09-09T12:00:00.00+01:00">
 </Result>
</ContentReferencingTable>
```

In the XML instance it can be seen that the Group CRID has two CRIDs associated with it, those of episode 1 and 2 of "Fox". In the example a DVB locator is used for episode one, the PDR already knows when and where this episode can be found. Episode two is somewhere in the future at an unknown time, so if the PDR tries to resolve that it will know to try again after the 9th of September 2001.

Note that the syntax of the locator is not specified here. For purposes of illustration, a locator has been dreamt up by appending an existing DVB locator with an "@" and a string to express time and duration according to ISO 8601 [8].

Acquire

The local storage management function will use any alternative locators to resolve recording conflicts. The chosen locator will then be used to tune to the specified channel at the specified time and record for the specified duration. To ensure that the content is recorded the system must monitor for changes in the location of the content. For example, the programme may be moved to a different channel. This may involve re-resolution of the CRID.

In addition, to accurately record the desired content it may be necessary to take advantage of lower-level system features such as Programme Delivery Control (PDC) or DVB event IDs. An example would be where the showing of a programme is delayed - if the original time and duration are followed the end of the programme will not be recorded.

Verification that actually the programme that was asked for has been recorded is not currently supported in *TV-Anytime*.

View

Once the episodes of the series have been acquired they are made available for viewing. As the viewer may want to view the associated metadata at the time of playback, the system should store the associated metadata at the time of selection or capture. If the metadata changes between selection and playback, it may be necessary to use version or timestamp information to present useful information to the user. For example, if one episode of a series advertised a particular guest actor as appearing, but did not take part, this may affect whether the user may wish to view the programme.

To allow users to know what they actually have recorded on their PDR, at least a minimal set of metadata needs to be kept with the content. In our example that could be ProgramInformation tables, allowing the user to see title and synopsis of programmes he recorded.

Finishing

This may involve a user preference system storing information about the viewing of this series or episode. This information could then be used by an agent to determine the preferences of the user. An extensive example of usage history can be found in annex A.

The following figure gives a graphic representation of this process.

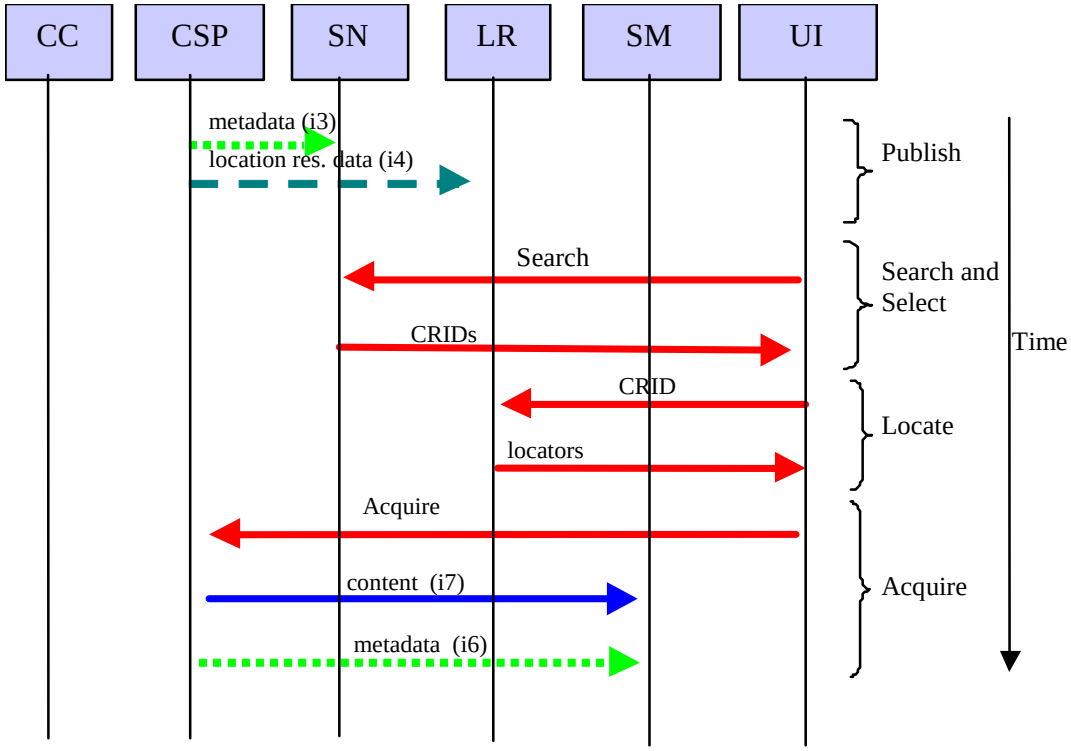

Figure 9: Dynamic behaviour of *TV-Anytime* system example

7.3 Example: Record highlights from a football match via an EPG

There are two ways to achieve this functionality. If the original match and the highlights are separate pieces of content identified by different CRIDs, the highlights can be captured by using the appropriate CRID. The alternative, described below, uses segmentation metadata.

Publish

A content service provider will publish a CRID that represents the football programme. The same or different service provider will publish content metadata that describes this programme. The same or different service provider will publish segmentation metadata that describes the highlights of this programme. The same or different service provider will publish location resolution data that describes where and when this programme may be acquired. The programme may be available from multiple content service providers.

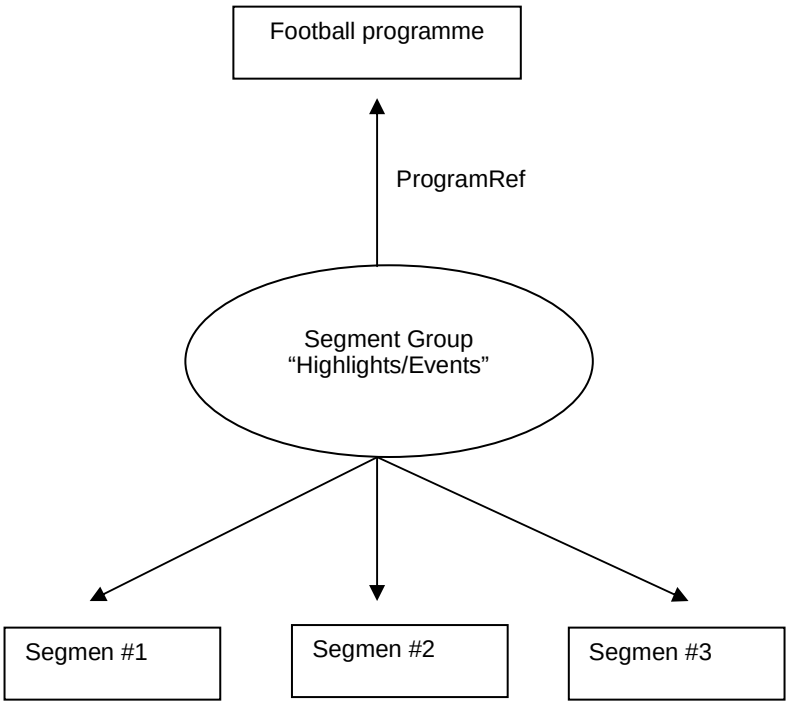

Figure 10: Highlight segmentation

The following XML snippet shows a minimal way to describe the Program Information and Program Location metadata for the football programme.

```
<ProgramDescription>
   <ProgramInformationTable>
      <ProgramInformation programId="crid://sport.com/football/match10">
         <BasicDescription>
            <Title type="main">Ireland Vs Saudi Arabia</Title>
            <Synopsis length="short">
               Ireland qualifies for the second round of the world cup
            </Synopsis>
         </BasicDescription>
      </ProgramInformation>
   </ProgramInformationTable>
   <ProgramLocationTable>
      <BroadcastEvent serviceIDRef="hbc10022311">
         <Program crid="crid://sport.com/football/match10" />
         <ProgramURL>dvb://1.4ee2.3f4/</ProgramURL>
         <PublishedStartTime>2002-06-05T18:00:00.00+01:00</PublishedStartTime>
         <PublishedDuration>PT6H</PublishedDuration>
      </BroadcastEvent>
   </ProgramLocationTable>
</ProgramDescription>
```

The segmentation metadata may be broadcast separately from the programme and it is associated programme information/programme location metadata. Segment metadata is "overlaid" on the original content. This means that the original content is published and various different segmentation schemes can be applied to it, for example, highlights with different durations. The following XML shows the segmentation metadata for the highlights.

The segment group is of type "Highlights/Events". It references the CRID of the football programme and contains references to three segments. The three highlight segments and the segment group are described as follows:

```
<ProgramDescription>
 <SegmentInformationTable>
  <SegmentList>
   <SegmentInformation segmentId="S27A67758-E714-4a4e-B994-
    3B650A443699">
    <ProgramRef crid="crid://sport.com/football/match10"/>
    <Description>
      <Title xml:lang="en">Highlight 1</Title>
      <Synopsis xml:lang="en">The first goal</Synopsis>
    </Description>
    <SegmentLocator>
     < MediaRelIncrTimePoint
       mediaTimeUnit="PT1N25F">10291
    </ MediaRelIncrTimePoint>
     < MediaIncrDuration
       mediaTimeUnit="PT1N25F">15470
     </ MediaIncrDuration>
    </SegmentLocator>
  </SegmentInformation>
  <SegmentInformation segmentId="S046C7C0F-BF83-4b4d-969E-
    204E8E82CF7C">
   <ProgramRef crid="crid://sport.com/football/match10"/>
   <Description>
    <Title xml:lang="en">Highlight 2</Title>
    <Synopsis xml:lang="en">The second goal</Synopsis>
   </Description>
   <SegmentLocator>
    < MediaRelIncrTimePoint mediaTimeUnit="PT1N25F">
     22291
    </ MediaRelIncrTimePoint>
    < MediaIncrDuration mediaTimeUnit="PT1N25F">
     26470
   </ MediaIncrDuration>
   </SegmentLocator>
  </SegmentInformation>
  <SegmentInformation segmentId="S5117353A-F598-4de1-968E-
    8C3D134C7642">
   <ProgramRef crid="crid://sport.com/football/match10"/>
    <Description>
     <Title xml:lang="en">Highlight 3</Title>
     <Synopsis xml:lang="en">The third goal</Synopsis>
    </Description>
    <SegmentLocator>
     < MediaRelIncrTimePoint mediaTimeUnit="PT1N25F">
       39291
    </ MediaRelIncrTimePoint>
     < MediaIncrDuration mediaTimeUnit="PT1N25F">
       55470
     </ MediaIncrDuration>
    </SegmentLocator>
  </SegmentInformation>
  </SegmentList>
  <SegmentGroupList>
   <SegmentGroupInformation groupId="G92F0C707-2ECB-403a-88FC-
    89EEF7961034">
    <ProgramRef crid="crid://sport.com/football/match10"/>
    <GroupType xsi:type="SegmentGroupTypeType"
      value="highlights/events"/>
    <Description>
     <Title xml:lang="en">Match Highlights</Title>
     <Synopsis xml:lang="en">Goals from the match</Synopsis>
    </Description>
    <Segments refList="S27A67758-E714-4a4e-B994-3B650A443699
            S046C7C0F-BF83-4b4d-969E-204E8E82CF7C
            S5117353A-F598-4de1-968E-8C3D134C7642"/>
   </SegmentGroupInformation>
```

```
  </SegmentGroupList>
 </SegmentInformationTable>
</ProgramDescription>
```

Search

The PDR will use the information from the ProgramInformation and ProgramLocation tables to render an EPG/ECG. The ProgramLocation table is used by the PDR to know the expected time and channel where the programmes will be shown, and the ProgramInformation table is used to provide the location-independent information about each programme (such as title and synopsis).

The user can then navigate around this EPG/ECG to find content that they wish to acquire.

Select

For our example we assume that the PDR will offer the user the option to view either the original football programme or the highlights of that programme via the EPG. The user wishes to watch the highlights and selects this option from the EPG.

```
EPG for channel XX
…..
6pm - 8pm Football: Ireland Vs Saudi Arabia | Highlights |
…..
```

As discussed in the View section the fact that the user has selected the highlights does not affect the actual content acquired. It simply indicates that the highlights segment metadata should also be acquired at some time and applied to the content when it is viewed.

At this point the usage history metadata table in the PDR could be updated, showing that the user has made a selection. An example XML snippet is shown in the first cookbook scenario.

This usage history could also be used by the PDR to fill the user preference metadata tables. As far as the user is concerned, the system will now autonomously make the content available at some point in the future.

Locate

Once the particular football programme has been chosen the programme CRID must be "resolved" to a unique locator (channel/time/duration in the broadcast case). This relies on the fact that the location resolution data is made available to the PDR. For this football programme there may be several locations, e.g. repeats. These locations contain the same content as far as the service provider is concerned.

For our example, the football programme has the following simple resolution tables associated with it:

```
<ContentReferencingTable>
 <!-- CRID resolution to locators -->
 <Result CRID="crid://sport.com/football/match10"
   status="resolved" complete="true" acquire="all">
   <LocationsResult>
    <Locator>dvb://1.4ee2.3f4;4f5@2001-04-
           05T21:00:00.00+01:00/PT00H45M
   </Locator>
   </LocationsResult>
 </Result>
</ContentReferencingTable>
```

In the XML instance it can be seen that the football programme has only one locator associated with its CRID. In the example a DVB locator is used. The PDR already knows when and where this episode can be found. Note that the syntax of the locator is not specified here. For purposes of illustration, a locator has been created by appending an existing DVB locator with an "@" and a string to express time and duration according to ISO 8601 [8].

Acquire

The locator will be used to tune to the specified channel at the specified time and record for the specified duration. To ensure that the content is recorded the system must monitor for changes in the location of the content. For example, the programme may be moved to a different channel. This may involve re-resolution of the CRID.

The segmentation metadata may be acquired at the same time as the content or it may be intermittently broadcast from the carousel and acquisition may occur at a later time.

There is also the possibility that the segmentation metadata could be updated over time. If the metadata changes between selection and playback it may be necessary to use version or timestamp information to present useful information to the user. For example, the highlights may be changed if one of the players in the match is later named as the best player of the tournament.

View

Once the football programme has been acquired they are made available for viewing. The user has captured the original football programme plus the segment metadata that applies to that programme. As the user chose the view the highlights from the EPG this is the version that should be displayed when the user views the programme. Of course as the original content has been captured an option can be provided to view the full programme as well.

How this preference (full content/highlights) is stored on the box is an implementation issue. It may be as simple as a flag attached to the content indicating whether the associated segment metadata is to be utilized.

As the user has selected the highlights, the following events occur when the programme is viewed:

- The `SegmentGroupInformation` is processed by the PDR. There are seen to be three segments associated with the CRID crid: //sport.com/football/match10.
- Each individual `SegmentInformation` is processed and the `segmentLocator` is used to index a particular segment of the referenced CRID.
- The segments are played. The PDR will play the highlights in a continuous ordered manner. Interstitial content such as title screens/advertising shall be part of the original programmes and identified as highlights/events.

To allow users to know what they actually have recorded on their PDR, at least a minimal set of metadata needs to be kept with the content. In our example this metadata could be the ProgramInformation tables, allowing the user to see the title and synopsis of programmes that have been recorded.

Finishing

This may involve a user preference system storing information about the viewing of this programme. An agent to determine the preferences of the user could then use this information. For example from this and other recordings it may be obvious that the user always views the highlights of sporting events and never the entire content.

7.4 Example: Select a particular showing of a programme from an EPG/ECG (in the broadcast case)

Publish

A content service provider will publish CRIDs for each programme in its schedule. They will also publish programme description and programme location metadata for these programmes. The same or different service provider will publish location resolution data that describes where and when the programmes from the schedule may be acquired.

The included XML snippets show an almost minimal way to describe this schedule. Two metadata tables are needed to describe the schedule. The ProgramInformation table contains information for each of the different programmes. The ProgramLocation table contains the time and channel information necessary to render an EPG. The link between the time and channel information for a programme and its location-independent description is made by the CRID. It is worth noting that the ProgramLocation table is *NOT* there to signal to the PDR where and when a particular programme really can be found: that is the job of content referencing and location resolution mechanism.

```
<ProgramDescription>
    <ProgramInformationTable>
      <ProgramInformation programId="crid://hbc.com/foxes/episode1">
         <BasicDescription>
            <Title type="main">
               The one where Fox jumps in the Potomac
            </Title>
            <Synopsis length="short">
               Fox goes to Washington and jumps in the Potomac
            </Synopsis>
         </BasicDescription>
      </ProgramInformation>
      <ProgramInformation programId="crid://hbc.com/news/six">
         <BasicDescription>
            <Title type="main">
               The HBC 6 o'Clock News
            </Title>
            <Synopsis length="short">
               The latest news and sports from around the world
            </Synopsis>
         </BasicDescription>
      </ProgramInformation>
      <ProgramInformation programId="crid://hbc.com/bear/woods">
         <BasicDescription>
            <Title type="main">
               The Bear Show in the Woods
            </Title>
            <Synopsis length="short">
               Bear sings a medley of songs from One Hundred Tree Wood
            </Synopsis>
         </BasicDescription>
      </ProgramInformation>
    </ProgramInformationTable>
    <ProgramLocationTable>
      <BroadcastEvent serviceIDRef = "hbc100022311">
         <Program crid="crid://hbc.com/news/six"/>
         <ProgramURL>dvb://1.4ee2.3f4/</ProgramURL>
         <PublishedStartTime>2002-06-05T18:00:00.00+01:00</PublishedStartTime>
         <PublishedDuration>PT30H</PublishedDuration>
      </BroadcastEvent>
      <BroadcastEvent serviceIDRef = "hbc100022311">
         <Program crid="crid://hbc.com/foxes/episode1"/>
         <ProgramURL>dvb://1.4ee2.3f4/</ProgramURL>
         <InstanceMetadataId>imi:fe11</InstanceMetadataId>
         <PublishedStartTime>2002-06-05T18:30:00.00+01:00</PublishedStartTime>
         <PublishedDuration>PT30H</PublishedDuration>
      </BroadcastEvent>
      <BroadcastEvent serviceIDRef = "hbc100022311">
         <Program crid="crid://hbc.com/bear/woods"/>
         <ProgramURL>dvb://1.4ee2.3f4/</ProgramURL>
         <PublishedStartTime>2002-06-05T19:00:00.00+01:00</PublishedStartTime>
         <PublishedDuration>PT60H</PublishedDuration>
      </BroadcastEvent>
   </ProgramLocationTable>
</ProgramDescription>
```

Insert instance metadata in the ProgramLocationTable to specify why the user might select that particular instance of the programme (e.g. edited for language).

Search

The PDR will use the information from the ProgramInformation and ProgramLocation tables to render an EPG/ECG. The ProgramLocation table is used by the PDR to know the expected time and channel where the programmes will be shown, and the ProgramInformation table is used to provide the location-independent information about each programme (such as title and synopsis).

The user can then navigate around this EPG/ECG to find content that they wish to acquire.

Select

Once the user has found something they wish to acquire, they have the choice of acquiring any instance of the programme, or the specific instance they chose from the EPG/ECG.

If the user selects "any instance" of the programme, the PDR uses the CRID from the Programme element of the BroadcastEvent table to start its acquisition process.

If the user selects the "this specific instance" of the programme, the PDR uses the CRID from the Programme element and the instance metadata identifier from the InstanceMetadataId element.

In the following example, we assume that the user has selected episode one of foxes (CRID "crid://hbc.com/foxes/episode1") and that they want this specific instance (instance metadata identifier "imi:fe1_1") rather than any other showing.

Locate

Given the CRID for the chosen programme, the location resolution functional unit will return a list of CRIDs that refer to showings of this programme. This relies on the fact that the location resolution data is made available to the box.

In the XML instance it can be seen that the CRID has two locators associated with it, those of the first showing and its repeat.

Note that the syntax of the locator is not specified here. For purposes of illustration, a locator has been created by appending an existing DVB locator with an "@" and a string to express time and duration according to ISO 8601 [8].

Acquire

The local storage management function could use any alternative locators to resolve recording conflicts when the user has not selected a preference for a specific instance. The chosen locator will then be used to tune to the specified channel at the specified time and record for the specified duration. When the user has selected a specific instance, the PDR will use the instance metadata identifier to decide which locator to use for acquisition.

Looking at the chosen example, the PDR would use the first locator in the content referencing result, because it has an instance metadata identifier of "imi:fe1_1".

To ensure that the content is recorded the system must monitor for changes in the location of the content. For example, the programme may be moved to a different channel. This may involve re-resolution of the CRID.

An interesting scenario to demonstrate is how a PDR can cope with changes in scheduling. Using the previous example, here is the content referencing result after a schedule change for the chosen episode of "Foxes".

```
<ContentReferencingTable>
<Result CRID="crid://hbc.com/foxes/episode1" status="resolved"
   complete="true" acquire="any">
  <LocationsResult>
   <Locator instanceMetadataId="imi:fe11">
     dvb://1.4ee2.3f4;4f5@2002-06-05T18:21:00.00+01:00/PT00H29M
    </Locator>
    <Locator instanceMetadataId="imi:fe12">
     dvb://1.4ee2.3f4;4f5@2002-06-08T21:30:00.00+01:00/PT00H29M
    </Locator>
  </LocationsResult>
 </Result>
</ContentReferencingTable>
```

In the above example, the first showing of episode one is delayed by twenty minutes. The PDR can still acquire the first showing because it can decide which locator to use by using the instance metadata identifier "imi:fe1_1".

The acquisition function of the PDR will need to use both the CRID ("crid://hbc.com/foxes/episode1") and the instance metadata identifier ("imi:fe1_1") to perform successful acquisition. This is because instance metadata identifiers are only unique within the scope of one CRID, so for example the "imi:fe1_1" instance metadata identifier might also be used with another CRID.

View

Once the chosen episode has been acquired it is made available for viewing. As the viewer may want to view the associated metadata at the time of playback, the system should store the associated metadata at the time of selection or capture. If the metadata changes between selection and playback, it may be necessary to use version or timestamp information to present useful information to the user.

To allow users to know what they actually have recorded on their PDR, at least a minimal set of metadata needs to be kept with the content. In our example that could be the ProgramInformation tables, allowing the user to see title and synopsis of programmes that have been recorded.

Finishing

This may involve a user preference system storing information about the viewing of this series or episode. This information could then be used by an agent to determine the preferences of the user. An extensive example of usage history can be found in annex A.

7.5 Example: Allow the user to select content from an on-demand content offer with associated pricing information, or seek lowest cost offer

Publish

A service provider will publish an on-demand content offer with associated pricing information. The offer is available via a broadcast channel (offer is pushed) or via an Internet server (offer is pulled).

For 500 Yen, this content can be played 5 times during 1 month as soon as content has been acquired. The commercial offer is valid during two month starting June, 1, 2004. Content is available for one month starting July, 1, 2004.

For 200 Yen, this content can be played only once.

The ImmediateViewing flag being set to "true", this content is subject to rights managements restrictions (see RMP).

```
<TVAMain xml:lang="ja" xmlns="urn:tva:metadata:2004" xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:tva:metadata:2004
schemas/tva_metadata_v13_am1.xsd">
    <ProgramDescription>
        <ProgramInformationTable>
            <ProgramInformation programId="crid://www.intergalactic.com/foxes">
                <BasicDescription>
                    <Title type="main">Foxes - The Movie</Title>
                    <Synopsis>Action-packed debut movie in which the Fox is elected governor of the
                state of California.</Synopsis>
                    <Keyword>Fox</Keyword>
                    <Genre href="urn:tva:metadata:cs:FormatCS:2004:3.5.7.3" type="main"/>
                </BasicDescription>
            </ProgramInformation>
        </ProgramInformationTable>
        <ProgramLocationTable>
            <OnDemandProgram>
                <Program crid="crid://www.intergalactic.com/foxes"/>
                <InstanceDescription>
                    <Title type="main">Foxes - The Movie</Title>
                    <PurchaseList>
                        <PurchaseItem start="2004-06-01T00:00:00" end="2004-07-31T00:00:00">
                            <Price currency="JPY">500</Price>
                            <Purchase>
                                <PurchaseType href="urn:tva:metadata:cs:PurchaseTypeCS:2004:playForPeriod"/>
                                <QuantityUnit href="urn:tva:metadata:cs:UnitTypeCS:2004:month"/>
                                <QuantityRange max="1"/>
                            </Purchase>
                            <Purchase>
                                <PurchaseType href="urn:tva:metadata:cs:PurchaseTypeCS:2004:playCounts"/>
                                <QuantityUnit href="urn:tva:metadata:cs:UnitTypeCS:2004:plays"/>
                                <QuantityRange max="5"/>
```

```
                        </Purchase>
                        <PricingServerURL>http://foxes.ondemand.com/prices/</PricingServerURL>
                    </PurchaseItem>
                </PurchaseList>
            </InstanceDescription>
            <PublishedDuration>P145M</PublishedDuration>
            <StartOfAvailability>2004-07-01T19:00:00.00+01:00</StartOfAvailability>
            <EndOfAvailability>2001-07-31T19:00:00.00+01:00</EndOfAvailability>
            <ImmediateViewing value="true"/>
        </OnDemandProgram>
        <OnDemandProgram>
            <Program crid="crid://www.intergalactic.com/foxes"/>
            <InstanceDescription>
                <Title type="main">Foxes - The Movie</Title>
                <PurchaseList>
                    <PurchaseItem start="2004-06-01T00:00:00" end="2004-07-31T00:00:00">
                        <Price currency="JPY">200</Price>
                        <Purchase>
                            <PurchaseType href="urn:tva:metadata:cs:PurchaseTypeCS:2004:playCounts"/>
                            <QuantityUnit href="urn:tva:metadata:cs:UnitTypeCS:2004:plays"/>
                            <QuantityRange max="1"/>
                        </Purchase>
                        <PricingServerURL>http://foxes.ondemand.com/prices/</PricingServerURL>
                    </PurchaseItem>
                </PurchaseList>
            </InstanceDescription>
            <PublishedDuration>P145M</PublishedDuration>
            <StartOfAvailability>2004-07-01T19:00:00.00+01:00</StartOfAvailability>
            <EndOfAvailability>2001-07-31T19:00:00.00+01:00</EndOfAvailability>
            <ImmediateViewing value="true"/>
        </OnDemandProgram>
    </ProgramLocationTable>
</ProgramDescription>
</TVAMain>
```

Search

If a user is looking all the offers under a given price ceiling for a particular movie (e.g. 1 000 Yens), he will either look at the prices proposed through the pushed offer, or retrieve additional information from the web server. The information will be returned as shown above. The user can then select the offer of his choice.

Locate, Acquire, View, Finishing

According to the TV-Anytime processes.

7.6 Example: Notify the user of something interesting based on their profile

Publish

A content service provider will publish a CRID that represents a programme. The same or different service provider will publish metadata that describes this programme. The same or different service provider will publish location resolution data that describes where and when the programme may be viewed.

In this example we will use a soccer game "World cup, Japan and Russia" which starts 4:30am EST June 14, 2002. The PDA is aware that the user has a preference for this type of content because of his profile that is stored on the PDR or NDR.

```
<ProgramDescription>
   <ProgramInformationTable>
      <ProgramInformation programId="crid://foo.com/soccer/worldcup/japan-russia">
         <BasicDescription>
            <Title type="main">
                 2002 FIFA World cup soccer game
            </Title>
            <Synopsis length="short">
```

```
            World cup soccer game, Japan versus Russia
          </Synopsis>
          <Keyword>World Cup</Keyword>
          <Keyword>soccer</Keyword>
          <Keyword>football</Keyword>
          <Keyword>Japan</Keyword>
          <Genre href = "urn:tva:metadata:cs:FormatCS:2004:3.2.3.12" type="main"/>
          <Genre href = "urn:tva:metadata:cs:FormatCS:2004:3.2.3" type="secondary"/>
        </BasicDescription>
        <MemberOf xsi:type="MemberOfType" crid="crid://foo.com/soccer/worldcup"/>
      </ProgramInformation>
   </ProgramInformationTable>
</ProgramDescription>
```

InstanceMetadata can be used to allow a PDR to build an EPG or to inform the user of the approximate schedule of an airing. The BroadcastEvent table in the instance metadata is used for that purpose. It is *NOT* there to signal to the PDR where and when a particular programme really can be found: that is the job of content referencing and location resolution mechanism. As stated before, the metadata is used mainly for attraction purposes. An example XML snippet of a BroadcastEvent table is given below:

```
<ProgramLocationTable>
   <BroadcastEvent serviceIDRef = "hbc100022311">
      <Program crid=" crid://foo.com/soccer/worldcup/japan-russia"/>
      <ProgramURL>dvb://1.4ee2.3f5/</ProgramURL>
         <PublishedStartTime>2001-04-05T21:00:00.00+01:00</PublishedStartTime>
         <PublishedDuration>PT30H</PublishedDuration>
      <Live value="false"/>
      <Repeat value="false"/>
      <FirstShowing value="false"/>
      <LastShowing value="false"/>
      <Free value="false"/>
   </BroadcastEvent>
</ProgramLocationTable>
```

Search

The FilteringAndSearchPreferences of the user's profile will be used by the PDR/NDR to filter and search for relevant programmes automatically. Every time the PDR/NDR gets a newer version of an EPG, or at other specified times, the PDR/NDR searches for programmes that matches the user's preferences. According to the results of this profile match, the PDR/NDR may issue a notification to the user. This notification may be issued via email, screen indication, or some other method that was selected by the user. The notification may include both programme and instance information. With this notification, the PDR may urge the user to perform some relevant action in the next phase.

```
<UserDescription>
   <UserPreferences>
      <mpeg7:UserIdentifier protected="true">
         <mpeg7:Name xml:lang="en">Jay</mpeg7:Name>
      </mpeg7:UserIdentifier>
      <mpeg7:FilteringAndSearchPreferences>
         <mpeg7:ClassificationPreferences preferenceValue="12">
            <mpeg7:Language>en</mpeg7:Language>
            <mpeg7:Genre href="urn:tva:metadata:cs:FormatCS:2004:3.2.3.12"/>
            <mpeg7:Subject>Japan</mpeg7:Subject>
         </mpeg7:ClassificationPreferences>
      </mpeg7:FilteringAndSearchPreferences>
   </UserPreferences>
</UserDescription>
```

Locate, Acquire, View, Finishing

Not relevant for this scenario.

7.7 Example: Personal Channel Service at my PDR

A viewer wants to set his/her week's watching schedule, after electronic guide information from many service providers has been stored at PDR. However he/she does not want to navigate all the programme information at programme guide application for setting watching schedule.

Also, the viewer wants PDR to generate the viewer's own schedule customized to his/her various preference or lifestyle. This generating procedure can be done at PDR automatically based on the viewer's usage history and user preference information.

Then the generated personal channel and its programme information provides following guide at programme guide application: at Monday 8:00 P.M. to 9:00 P.M., CNN news, and at 9:00 to 10:00 P.M., special action movie from HBO, etc. Surely, this re-arranged programme schedule combined from many broadcasting providers builds into one personal channel.

The personal channel at PDR provides above rescheduled programmes by a user's preference at the user's preferred date on the personal channel.

For this personal channel at PDR, the following operations must be done by a user's PDR in sequence order.

- A user's usage history is stored.
- The user's preference for date (day and time), genre per date, and programme title is extracted by analyzing the user's usage history.
- A new channel for the user is generated.
- Programmes are determined to be broadcast on the personal channel at the user's preferred date by the user's preference information.
- A new personalized InstanceDescriptionMetadata is generated for informing the user that a new programme instance is included in the personal channel.

When this personal channel service is compared with an existent preference-based-EPG service, the object of personal channel service is to provide a new channel that broadcasts newly scheduled programmes according to user's preference, but the object of preference-based-EPG service is to make user find easily his/her preferred programmes among the enormous programmes on the EPG.

That is, the personal channel service relocates programmes by a user's title and date preference and provides the programmes only on the personal channel, but the preference-based-EPG service just differently represents the programme lists of all channels according to the degree of user preference.

For a general understanding about the personal channel service, figure 11 shows a conceptual drawing of personal channel service. A common service provider generates and provides Instance Description Metadata to PDRs. Then a PDR uses this Instance Description Metadata to render the EPG for informing a user of the schedule of usually broadcast programmes. In addition to the usage, in the personal channel service, Personal Channel Controller in the PDR uses the Instance Description Metadata for choosing user preferred programmes for personal channel, and then stores new instances of the selected programmes in the personalized instance description metadata.

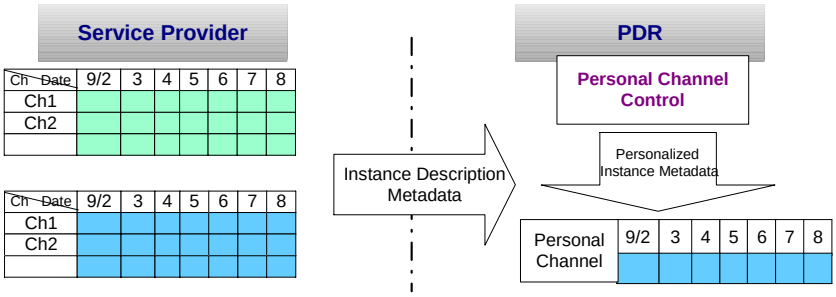

Figure 11: Conceptual drawing of Personal Channel

In detail, inter-operations between a service provider and a user for the personal channel service are shown in figure 12. As explained above, a service provider offers CRID, location resolution data and metadata such as ContentDescriptionMetadata (including ProgramInformation, GroupInformation), InstanceDescriptionMetadata (including ProgramLocation, ServiceInformation). A user's PDR simply can render EPG using the metadata, and then the user can choose and watch a programme when the programme has been already scheduled by the service provider. But in the personal channel service, the Personal Channel Controller in the PDR relocates programmes according to user preference on the personal channel. And then the new schedule of programmes for the personal channel is informed to users by including new instances in the personalized instance metadata.

In the following, we explain how the personal channel service may work using the Content Referencing specification (S-4) and Metadata specification (S-3). We follow all the phases identified in a *TV-Anytime* System, e.g. Publish, Search, Select, Locate, Acquire, View, and Finish. In the personal channel service, a new phase, PDRs Search and Select, is included after the Publish phase because the PDR rearranges programmes provided by service providers and recreate Instance Description Metadata locally for the virtual channel.

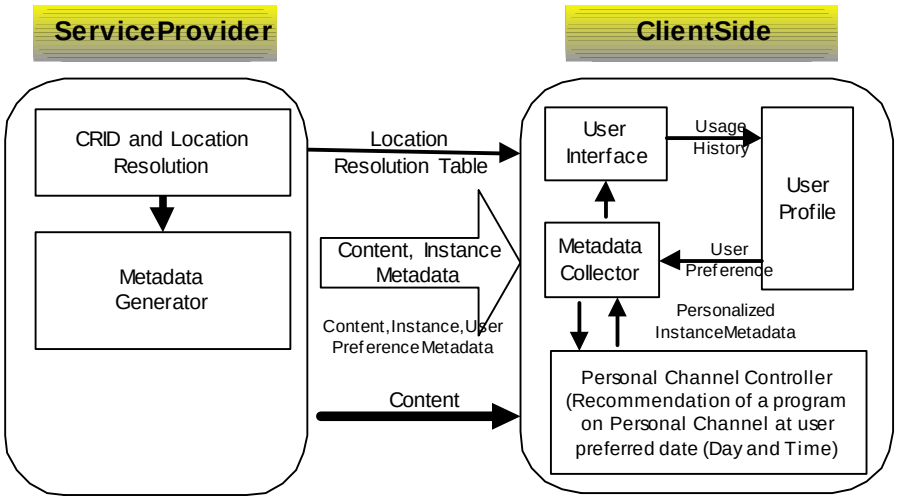

Figure 12: Interaction between Service Provider and Client

Publish

A content service provider will publish a CRID that represents a programme, the same or different service provider will publish location resolution data that describes where and when the programme may be viewed, and the same or different service provider will publish programme description metadata and programme instance metadata.

For the personal channel service, ContentDescriptionMetadata (including ProgramInformation, GroupInformation) and InstanceDescriptionMetadata (including ProgramLocation, ServiceInformation) must be offered by the service provider, and UserPreference metadata must be provided by the PDR, because Personal Channel Controller in the PDR chooses programmes which will be broadcast on personal channel by programme description metadata including programme group information, programme instance metadata, and user preference data.

Of the three kinds of metadata, the first metadata, ContentDescriptionMetadata (including ProgramInformation, GroupInformation) is general information about a piece of content that does not change regardless of how the content is published or broadcast.

The following XML snippets show GroupInformation element of a Korean Drama, "Sangdo" which has several episodes, and a few ProgramInformation elements that contain information for each episode of the group, "Sangdo".

```
<ProgramDescription>
  <ProgramInformationTable>
    <ProgramInformation programId="crid://imbc.com/sangdo/episode17">
      <BasicDescription>
        <Title> Sangdo_17 </Title>
        <Synopsis> The one where Sangdo is embarrassed with the rumour</Synopsis>
        <Genre href = "urn:tva:metadata:cs:FormatCS:2004:3.4.2.1" type="main"/>
      </BasicDescription>
    </ProgramInformation>
```

```
      <ProgramInformation programId="crid://imbc.com/sangdo/episode18">
         <BasicDescription>
            <Title> Sangdo_18 </Title>
            <Synopsis> The one where Sangdo sold his clothes cheaply </Synopsis>
            <Genre href = "urn:tva:metadata:cs:FormatCS:2002:3.4.2.1" type="main"/>
         </BasicDescription>
      </ProgramInformation>
   </ProgramInformationTable>
   <GroupInformationTable>
      <GroupInformation groupId="crid://imbc.com/sangdo/several" ordered="true"
numOfItems="2">
         <GroupType xsi:type="ProgramGroupTypeType" value="series"/>
         <BasicDescription>
            <Title>Sangdo</Title>
            <Synopsis>Several episodes of Sangdo</Synopsis>
            <Genre href = "urn:tva:metadata:cs:FormatCS:2004:3.4.2.1" type="main"/>
         </BasicDescription>
      </GroupInformation>
   </GroupInformationTable>
</ProgramDescription>
```

The second metadata, InstanceDescriptionMetadata is also offered by the service provider. Instance DescriptionMetadata (including ProgramLocationTable) can be used to allow a PDR to build an EPG or to inform the user of the approximate schedule of an airing. The BroadcastEvent element in the instance description metadata is used for that purpose.

For the personal channel service, the InstanceDescriptionMetadata plays a great role in addition to the general EPG function. InstanceDescriptionMetadata is referred in PDR's selecting a programme for a personal channel according to the user preference and used in announcing newly included channel and programme instances to the user. That will be explained in more detail in the phrase of PDR's Search and Select.

The followed XML snippet shows ProgramLocation table that holds information of each programme instance and ServiceInformation table that contains channel information such as service id, service name, owner, and so on.

```
<ProgramDescription>
   <ProgramLocationTable>
      <BroadcastEvent serviceIDRef="Ch1">
         <Program crid="crid://imbc.com/sangdo/episode17"/>
         <ProgramURL>dvb://1.4ee2.3f4/</ProgramURL>
         <PublishedStartTime>2002-04-22T22:00:00</PublishedStartTime>
         <PublishedDuration>PT50M</PublishedDuration>
      </BroadcastEvent>
      <BroadcastEvent serviceIDRef="Ch1">
         <Program crid="crid://imbc.com/sangdo/episode18"/>
         <ProgramURL>dvb://1.4ee2.3f4/</ProgramURL>
         <PublishedStartTime>2002-04-23T22:00:00</PublishedStartTime>
         <PublishedDuration>PT50M</PublishedDuration>
      </BroadcastEvent>
</ProgramLocationTable>
<ServiceInformationTable>
   <ServiceInformation serviceId="Ch1">
      <Name>MBC</Name>
         <Owner>MBC</Owner>
   </ServiceInformation>
</ServiceInformationTable>
</ProgramDescription>
```

The third metadata, UserPreference metadata is automatically generated by User Profile in the PDR and it is an essential information to determine an airing date and programme on the personal channel. UserPreference metadata for the personal channel service uses minimal preference information such as title preference and genre preference per date.

The followed XML snippet shows FilteringAndSearchPreferences that specifies a user's filtering and/or searching preference for audio-visual content. These preferences are specified by creation-, classification-related properties of the content.

```
<UserDescription>
   <UserPreferences>
      <mpeg7:UserIdentifier>
         <mpeg7:Name xml:lang="en">etri</mpeg7:Name>
      </mpeg7:UserIdentifier>
      <mpeg7:FilteringAndSearchPreferences>
         <mpeg7:CreationPreferences>
            <mpeg7:Title preferenceValue="99">sangdo</mpeg7:Title>
         </mpeg7:CreationPreferences>
      </mpeg7:FilteringAndSearchPreferences>
      <mpeg7:FilteringAndSearchPreferences>
         <mpeg7:ClassificationPreferences>
            <mpeg7:Genre href = "urn:tva:metadata:cs:FormatCS:2004:3.4.2.1"
preferenceValue="80"/>
         </mpeg7:ClassificationPreferences>
         <mpeg7:PreferenceCondition>
            <mpeg7:Time recurrence="weekly">
               <mpeg7:TimePoint>2002-04-23T11:00</mpeg7:TimePoint>
               <mpeg7:Duration>PT1H</mpeg7:Duration>
               </mpeg7:Time>
         </mpeg7:PreferenceCondition>
      </mpeg7:FilteringAndSearchPreferences>
   </UserPreferences>
</UserDescription>
```

PDR's Search and Select

After the three kinds of metadata are offered, Personal Channel Controller in the PDR makes a new channel as personal channel with the exception of usual broadcast, cable, and satellite channels. Personal Channel Controller also selects programmes that are going to be broadcast on personal channel by user's preference data, programme description metadata, and programme instance metadata.

For this programme selection process, the following operations must be done by the Personal Channel Controller.

- First, a user's preferred date (day and time) is checked.
- Second, which genre is preferred at the date is checked.
- Third, a programme that is included in the preferred genre and has higher title preference is chosen as a new programme for personal channel.
- Fourth, the information about a newly included channel and its new instance is included in the InstanceDescriptionMetadata (ProgramLocation) to announce to the user.

At fourth step of above operations, a set of new service information and broadcast event information are generated and added into InstanceDescriptionMetadata.

Note that the location resolving procedure of programmes on personal channel is tightly related to the instance identification schemes and related descriptions - Content Referencing Table and Program Location Table - generated from the PDR. To resolve the location of a certain instance of a programme, the *TV-Anytime* Forum supports content referencing by CRID. By content referencing scheme, the PDR generates a new locator and assigns it for a newly scheduled programme. Then, the location of a certain programme is identified by a certain CRID.

The included XML snippet shows an original instance of the selected programme scheduled in the "MBC" channel and a newly included channel, that is "PERSONAL", and its new instance.

```
<ProgramDescription>
   <ProgramLocationTable>
      <BroadcastEvent serviceIDRef="Ch1">
         <Program crid="crid://imbc.com/sangdo/episode18"/>
         <ProgramURL>dvb://1.4ee2.3f4/</ProgramURL>
         <PublishedStartTime>2002-04-16T22:00:00.00</PublishedStartTime>
         <PublishedDuration>PT50M</PublishedDuration>
      </BroadcastEvent>
      <BroadcastEvent serviceIDRef="Ch2">
         <Program crid="crid://imbc.com/sangdo/episode18"/>
```

```
        <ProgramURL> My_PDR/personal/</ProgramURL>
        <PublishedStartTime>2002-04-20T11:00:00.00</PublishedStartTime>
        <PublishedDuration>PT50M</PublishedDuration>
      </BroadcastEvent>
   </ProgramLocationTable>
   <ServiceInformationTable>
      <ServiceInformation serviceId="Ch1">
        <Name>MBC</Name>
        <Owner>MBC</Owner>
      </ServiceInformation>
      <ServiceInformation serviceId="Ch2">
        <Name>PERSONAL</Name>
        <Owner>My_PDR</Owner>
      </ServiceInformation>
   </ServiceInformationTable>
</ProgramDescription>
```

User's Search

After the PDR's search and select phase, PDR can use ProgramInformation metadata and newly generated ProgramLocation metadata to render an EPG/ECG on the User Interface in PDR. The EPG/ECG also represents the personal channel information like other channels.

Next to the EPG/ECG generation, the user can navigate around this EPG/ECG and consume audiovisual contents. This audiovisual content consumption history for a user is described in the UsageHistory as lists of the actions performed by the user over an observation period, which can subsequently be used by the User Profile in PDR to generate user preferences.

As the above general case, when a user surveys the programme information and group information of a specific programme, that affects the user's preference positively because this survey presents the user's interest in the programme. For this positive effect on the user's preference, a new UserAction item is included in the UsageHistory in the User Profile, and then the UsageHistory is used by the User Profile to fill the user preference metadata table which specially includes preference for programme title, date, and genre per date.

The followed XML snippet shows UsageHistory that contains UserAction item whose type is ViewGuide.

```
<UserDescription>
   <UsageHistory id="usage-history-001" allowCollection="true">
      <mpeg7:UserIdentifier protected="true">
           <mpeg7:Name xml:lang="en">etri</mpeg7:Name>
      </mpeg7:UserIdentifier>
      <mpeg7:UserActionHistory protected="false">
         <mpeg7:ObservationPeriod>
              <mpeg7:TimePoint>2002-04-18T10:00</mpeg7:TimePoint>
              <mpeg7:Duration>PT3H</mpeg7:Duration>
         </mpeg7:ObservationPeriod>
         <mpeg7:UserActionList id="ua-list-001" numOfInstances="1" totalDuration="PT20M">
            <mpeg7:ActionType href="urn:mpeg:mpeg7:cs:ActionTypeCS:2004:3.4">
               <mpeg7:Name>ViewGuide</mpeg7:Name>
            </mpeg7:ActionType>
            <mpeg7:UserAction>
               <mpeg7:ActionTime>
                  <mpeg7:MediaTime>
                     <mpeg7:MediaTimePoint>2002-04-18T10:05:00</mpeg7:MediaTimePoint>
                     <mpeg7:MediaDuration>PT00H20M</mpeg7:MediaDuration>
                  </mpeg7:MediaTime>
               </mpeg7:ActionTime>
               <mpeg7:ProgramIdentifier organization="TVAF"
type="CRID">crid://imbc.com/sangdo/several </mpeg7:ProgramIdentifier>
            </mpeg7:UserAction>
         </mpeg7:UserActionList>
      </mpeg7:UserActionHistory>
   </UsageHistory>
</UserDescription>
```

In the User's Search phase, the user is more willing to choose an instance for the personal channel, because the personal channel provides a user preferred programme at the preferred date using the UserPreference information explained above.

User's Selection

During a user's navigation, if the user has found a programme that the user wishes to acquire from the personal channel, the user selects the programme on the EPG/ECG.

The specific instance from personal channel is selected using both the CRID from the Programme element and the instance metadata identifier from the InstanceMetadataId element. This is because instance metadata identifiers are only unique within the scope of one CRID.

In this personal channel scenario, we assume that a user has selected the 18th episode of Sangdo, a Korean drama whose CRID is "crid://imbc.com/sangdo/episode18", and that the user wants this specific instance from personal channel(instance metadata identifier "imi:my_PDR/etri2") rather than any other showing.

Locate

Given the CRID for the chosen programme, the location resolution functional unit will return a list of locators that refer to showings of this programme. This relies on the fact that the location resolution data is made available to the box.

In the following XML instance, it can be seen that the CRID has two locators associated with it, those of the general DVB and personal channel.

```
<ContentReferencingTable>
   <Result CRID="crid://imbc.com/sangdo/episode18"
status="resolved" complete="true" acquire="any">
   <LocationsResult>
      <Locator>
           dvb://1.4ee2.3f4;4f5@2002-04-16T22:00:00.00/PT00H50M
        </Locator>
        <Locator>
My_PDR/personal/sangdo18@2002-04-20T11:00:00.00/PT00H50M
        </Locator>
     </LocationsResult>
   </Result>
</ContentReferencingTable>
```

Acquire

As explained in the User's Select phase, because the user has selected a specific instance from the personal channel, the PDR would use the second locator in the content referencing result. And the chosen locator will then be used to tune to the specified channel at the specified time and record for the specified duration. The PDR will use the instance metadata identifier to decide which locator to use for acquisition.

Looking at the chosen example, the PDR would use the second locator in the content referencing result, because it has an instance metadata identifier of "imi:my_PDR/etri2". And the chosen locator will then be used to tune to the specified channel at the specified time and record for the specified duration.

View

Once the chosen episode has been acquired it is made available for viewing.

Finish

In this phase, storing information about the viewing of this series or episode in the usage history data is done by PDR.

7.8 Example: Programme made up of segments from multiple providers and maintain the latest news on my PDR

For this example, there are two cases:

- A programme has been pre-recorded and is delivered at a scheduled time and associated with a group CRID. This programme is made of several programmes having their own crid federated under the umbrella of the overall programme groupCrid. The user-personalized service knows the programme elements of the user interest, which crids will be used to allow the capture and update of the desired programme elements.
- A live CNN or Eurosport programme each identified by a CRID is being transmitted and recorded in its entirety. Segmentation information is provided after the programme has ended that will allow the user to automatically access the segments of his interest has previously defined as part of his subscription profile.

This scenario explores the second case.

A viewer has subscribed to a "latest, personalized news" service from their provider. They want their world news from CNN, and their sports news from Eurosport. They expect to be able to view only the news requested from each of these providers be captured and compiled into a virtual programme that begins with CNN then automatically goes to Eurosport. They also require that they can jump between items using their remote control and if required view a list of the available segments of news in a list format- that describes the content in each clip.

They have also requested that the business news segments be updated at every available opportunity, which means that each bulletin will be recorded in its entirety. The PDR may be configured to remove content that is not anymore used.

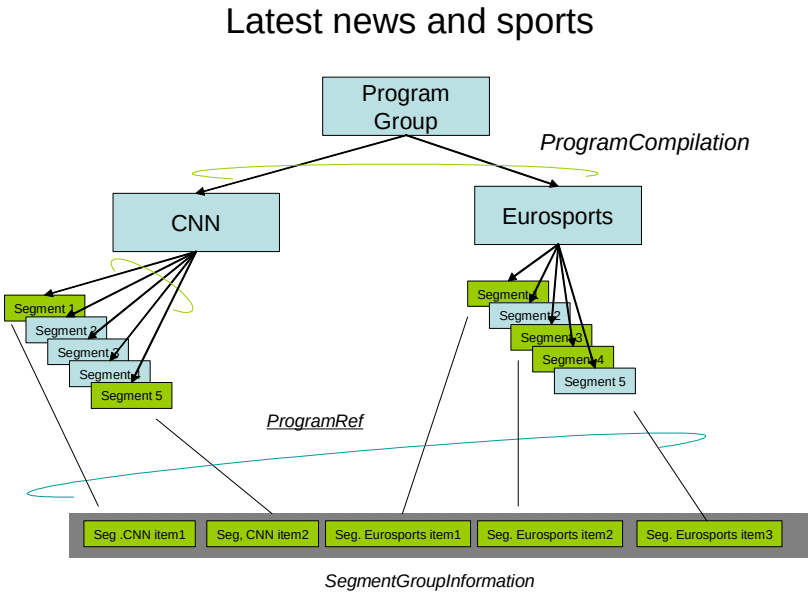

Figure 13: Compilation of updated segments from multiple providers

Publish

A content service provider will publish a CRID that represents a programme. The same or a different service provider will publish metadata that describes this programme. The same or a different service provider will publish location resolution data that describes where and when the programme may be viewed.

The included XML snippets show an almost minimal way to describe this schedule. Two metadata tables are needed to describe the schedule. The ProgramInformationTable contains information for each of the different programmes. It is worth noting that the ProgramLocation table is *NOT* there to signal to the PDR where and when a particular programme really can be found: that is the task of content referencing and location resolution mechanism.

In this example we will use CNN latest news and Eurosport latest sport. The PDR is aware that the user has a preference for this type of content because of his latest news subscription profile that is stored on the PDR or NDR. Both CNN and Eurosport programmes are described as ProgramCompilation in the GroupInformationTable.

```
<TVAMain xml:lang="en"
   publisher="TVA"
   publicationTime="2002-08-02T09:30:47-05:00"
   rightsOwner="TVA"
   version="0">

 <CopyrightNotice>TVA</CopyrightNotice>
 <ProgramDescription>
   <ProgramInformationTable>
      <ProgramInformation programId="crid://cnn.com/latestnews">
         <BasicDescription>
            <Title type="main">CNN Business update</Title>
            <Synopsis length="short">
                 Latest version of the CNN business news
            </Synopsis>
         </BasicDescription>
      </ProgramInformation>
      <ProgramInformation programId="crid://eurosport.com/latest">
         <BasicDescription>
            <Title type="main">Eurosport sport update</Title>
            <Synopsis length="short">
                 A collection of the latest sport news by Eurosport
            </Synopsis>
         </BasicDescription>
      </ProgramInformation>
   </ProgramInformationTable>
   <GroupInformationTable>
      <GroupInformation groupId="crid://foo.bar.com/gary" ordered="true" numOfItems="2">
         <GroupType xsi:type="ProgramGroupTypeType" value="programCompilation"/>
         <BasicDescription>
            <Title type="main">
               Gary's latest business and sports news
            </Title>
            <Synopsis length="short">
               Mix from CNN business news and Eurosport sports
            </Synopsis>
         </BasicDescription>
      </GroupInformation>
   </GroupInformationTable>
 </ProgramDescription>
</TVAMain>
```

The segmentation metadata will be broadcast after each programme and it is associated programme information/programme location metadata. The following XML snippet shows the segmentation metadata for the latest news and sports. It references the CNN latest news programme, which contains references to two segments and the Eurosport latest sport programme, which contains references to three segments. The two and three latest segments and the segment group are described as follows:

```
<TVAMain xml:lang="en"
   publisher="TVA"
   publicationTime="2002-08-02T09:30:47-05:00"
   rightsOwner="TVA"
   version="0">
   <CopyrightNotice>TVA</CopyrightNotice>
   <ProgramDescription>
      <SegmentInformationTable>
         <SegmentList>
            <SegmentInformation segmentId="S1ff-efa5-e567-12ff">
               <ProgramRef crid="crid://cnn.com/latestnews"/>
               <Description>
                  <Title xml:lang="en">CNN Latest Business News</Title>
                  <Synopsis xml:lang="en">Fox goes to the stockmarket</Synopsis>
               </Description>
               <SegmentLocator>
                  < MediaRelIncrTimePoint mediaTimeUnit="PT1N25F">13777</
MediaRelIncrTimePoint>
                  < MediaIncrDuration mediaTimeUnit="PT1N25F">16780</ MediaIncrDuration>
               </SegmentLocator>
```

```
          </SegmentInformation>
          <SegmentInformation segmentId="S2ff-efa5-e567-34ff">
             <ProgramRef crid="crid://cnn.com/latestnews"/>
             <Description>
                <Title xml:lang="en">CNN Latest Business News</Title>
                <Synopsis xml:lang="en">Interview with Fox</Synopsis>
              </Description>
              <SegmentLocator>
                < MediaRelIncrTimePoint mediaTimeUnit="PT1N25F">27890</
MediaRelIncrTimePoint>
                < MediaIncrDuration mediaTimeUnit="PT1N25F">25456</ MediaIncrDuration>
              </SegmentLocator>
          </SegmentInformation>
          <SegmentInformation segmentId="S5ff-efa5-e567-12ff">
             <ProgramRef crid="crid://eurosport.com/latest"/>
             <Description>
                <Title xml:lang="en">Eurosport Latest Sports</Title>
                <Synopsis xml:lang="en">
                   Fox wins the tennis at the Paris Open
                </Synopsis>
             </Description>
             <SegmentLocator>
                   < MediaRelIncrTimePoint mediaTimeUnit="PT1N25F">
                   12901
                   </ MediaRelIncrTimePoint>
                < MediaIncrDuration mediaTimeUnit="PT1N25F">
                   15470
                </ MediaIncrDuration>
             </SegmentLocator>
           </SegmentInformation>
           <SegmentInformation segmentId="S5ff-efa5-e567-34ff">
             <ProgramRef crid="crid://eurosport.com/latest"/>
             <Description>
                <Title xml:lang="en">Eurosport Latest Sports</Title>
                <Synopsis xml:lang="en">
                   Fox beats Tiger at Augusta.
                </Synopsis>
             </Description>
             <SegmentLocator>
                   < MediaRelIncrTimePoint mediaTimeUnit="PT1N25F">
                   22291
                   </ MediaRelIncrTimePoint>
                   < MediaIncrDuration mediaTimeUnit="PT1N25F">
                   26470
                   </ MediaIncrDuration>
             </SegmentLocator>
           </SegmentInformation>
           <SegmentInformation segmentId="S5ff-ee44-e567-34ff">
             <ProgramRef crid="crid://eurosport.com/latest"/>
                <Description>
                <Title xml:lang="en">Eurosport Latest Sports</Title>
                <Synopsis xml:lang="en">
                   Fox loses sumo championship.
                </Synopsis>
             </Description>
             <SegmentLocator>
                < MediaRelIncrTimePoint mediaTimeUnit="PT1N25F">
                   25291
                </ MediaRelIncrTimePoint>
                < MediaIncrDuration mediaTimeUnit="PT1N25F">
                   27885
                </ MediaIncrDuration>
             </SegmentLocator>
          </SegmentInformation>
       </SegmentList>
       <SegmentGroupList>
          <SegmentGroupInformation groupId="G234-ac4a-dddd-ffa9">
             <ProgramRef crid="crid://foo.bar.com/gary"/>
             <GroupType xsi:type="SegmentGroupTypeType" value="tableOfContents"/>
```

```
            <Description>
               <Title xml:lang="en">
                  Items from CNN news and Eurosport
               </Title>
               <Synopsis xml:lang="en">
                  Segment group containing segments from CNN and Eurosport
               </Synopsis>
            </Description>
            <Groups refList="G111-4444-ffff-eeee G222-4444-ffff-eeee"/>
         </SegmentGroupInformation>
        <SegmentGroupInformation groupId="G111-4444-ffff-eeee">
           <ProgramRef crid="crid://cnn.com/latestnews"/>
           <GroupType xsi:type="SegmentGroupTypeType" value="tableOfContents"/>
           <Description>
        <Title xml:lang="en">Items from CNN news</Title>
        <Synopsis xml:lang="en">
           Segment group containing segments from CNN
        </Synopsis>
       </Description>
       <Segments refList="S12ff-efa5-e567-12ff
                 S12ff-efa5-e567-34ff"/>
     </SegmentGroupInformation>
     <SegmentGroupInformation groupId="G222-4444-ffff-eeee">
       <ProgramRef crid="crid://eurosport.com/latestsports"/>
       <GroupType xsi:type="SegmentGroupTypeType" value="tableOfContents"/>
       <Description>
          <Title xml:lang="en">Items from Eurosport</Title>
          <Synopsis xml:lang="en">
             Segment group containing segments from Eurosport
          </Synopsis>
       </Description>
       <Segments refList="S5ff-efa5-e567-12ff
                  S5ff-efa5-e567-34ff
                   S5ff-ee44-e567-34ff"/>
     </SegmentGroupInformation>
    </SegmentGroupList>
   </SegmentInformationTable>
  </ProgramDescription>
</TVAMain>
```

Search

Every time the PDR/NDR gets a newer version of an EPG, or at other specified times, the PDR/NDR searches for the appropriate programmes. According to the results of this profile match, the PDR/NDR may capture the latest versions of the programmes and may also remove obsolete versions.

View

Once the latest versions of the programmes and the associated segmentation information have been acquired it is made available for viewing. As the viewer may want to view the associated metadata at the time of playback, the system should store the associated metadata. To allow users to know what they actually have recorded on their PDR, at least a minimal set of metadata needs to be kept with the content. In our example that could be the ProgramInformationTable, allowing the user to see title and synopsis of programmes that have been recorded, and the SegmentInformationTable, allowing the use to see title and synopsis of the segments.

Acquire

The CRIDs of the programmes corresponding to the user subscription service must be "resolved" to a unique locator (channel/time/duration). (Similar to previous scenarios).

Finish

Not relevant for this scenario.

7.9 Example: Usage Scenarios for Bi-directional Metadata Transport

This example is for providing a usage scenario for bi-directional metadata transport using the existing *TV-Anytime* specification. HTTP and TCP/IP are used for protocols of metadata transport on IP network.

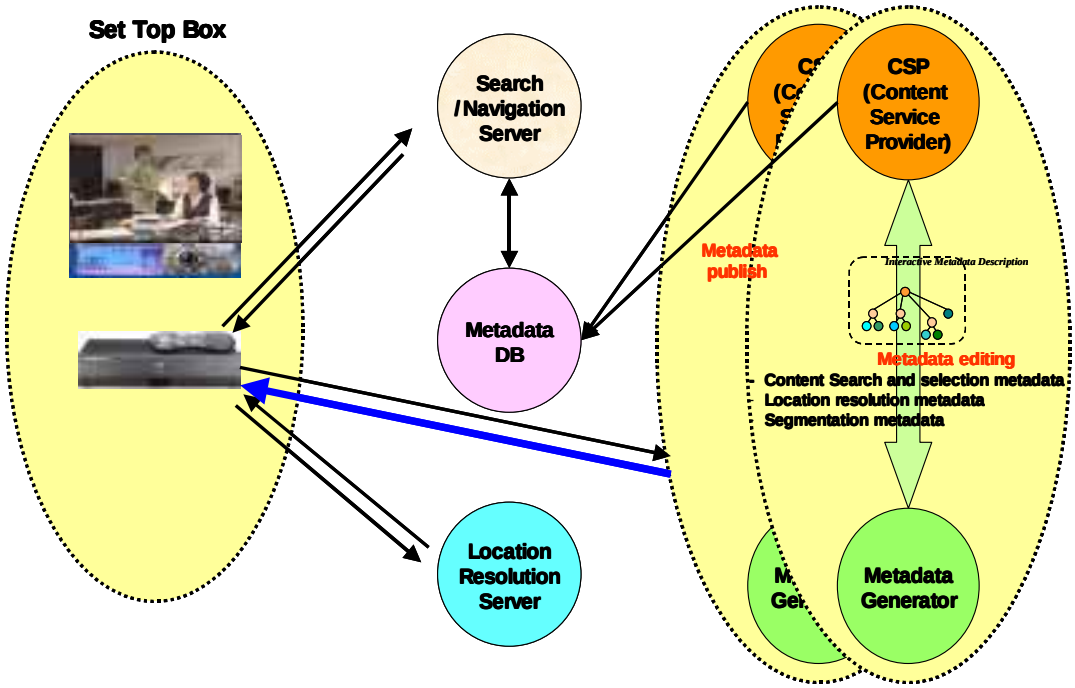

Figure 14: Bi-directional metadata transport

Publish

A content service provider will publish a CRID that represents a programme. The same or different service provider will publish metadata that describes this programme to metadata DB server. The metadata server aggregates and edits the metadata for establishing a metadata DB.

Search

A query can be constructed to get programme information from the metadata server.

The query might be:

```
<get_Data>
 <QueryConstraints>
  <PredicateBag type='AND'>
  <BinaryPredicate fieldID='Genre' fieldValue='Fiction' />
  <BinaryPredicate fieldID='Keyword' fieldValue='separation' />
 </PredicateBag>
 </QueryConstraints>
 <RequestedTables>
  <Table type='ProgramInformationTable'>
   <SortCriteria fieldID='tvaf:Title' />
  </Table>
  <Table type='ProgramLocationTable' />
 </RequestedTables>
</get_Data>
```

Below is a possible result of such a query.

```
<ProgramDescription>
   <ProgramInformationTable>
      <ProgramInformation programId="crid://www.keti.re.kr/2002002">
         <BasicDescription>
            <Title type="seriesTitle">Ghost mamma</Title>
            <Synopsis> A year later, Jiseok attempts to commit suicide, because he has
been feeling guilty and lonely.</Synopsis>
            <Keyword>love story</Keyword>
            <Keyword>separation</Keyword>
            <Keyword>death</Keyword>
            <Genre href="urn:tva:metadata:cs:ContentCS:2004:5.1" type="main"/>
            <Genre href="urn:tva:metadata:cs:FormatCS:2004:3.1" type="secondary"/>
            <ParentalGuidance>
               <mpeg7:ParentalRating href="urn:mpeg:mpeg7:cs:MPAAParentalRatingCS:G">
                  <mpeg7:Name>G</mpeg7:Name>
               </mpeg7:ParentalRating>
               <mpeg7:Region>UK</mpeg7:Region>
            </ParentalGuidance>
            <Language type="original">ko</Language>
            <CreditsList>
               <CreditsItem role="urn:mpeg:mpeg7:cs:MPEG7RoleCS:ANCHOR">
                  <PersonNameIDRef ref="PN61"/>
               </CreditsItem>
               <CreditsItem role="urn:mpeg:mpeg7:cs:MPEG7RoleCS:ACTRESS">
                  <PersonNameIDRef ref="PN15"/>
               </CreditsItem>
            </CreditsList>
            <ProductionDate>
               <TimePoint>2002</TimePoint>
            </ProductionDate>
            <ProductionLocation>ko</ProductionLocation>
            <CreationCoordinates>
               <CreationDate>
                  <TimePoint>2002-03-21</TimePoint>
               </CreationDate>
               <CreationLocation>ko</CreationLocation>
            </CreationCoordinates>
            <ReleaseInformation>
               <ReleaseDate>
                  <DayAndYear>2002-08-11</DayAndYear>
               </ReleaseDate>
               <ReleaseLocation>ko</ReleaseLocation>
            </ReleaseInformation>
         </BasicDescription>
      </ProgramInformation>
   </ProgramInformationTable>
   <ProgramLocationTable>
      <BroadcastEvent>
         <Program crid="crid://www.keti.re.kr/2002002"/>
         <ProgramURL>null</ProgramURL>
         <PublishedStartTime>2002-08-01T16:41:00</PublishedStartTime>
         <PublishedDuration>P55M</PublishedDuration>
         <Live value="false"/>
         <Repeat value="false"/>
         <FirstShowing value="true"/>
         <LastShowing value="false"/>
         <Free value="true"/>
      </BroadcastEvent>
   </ProgramLocationTable>
</ProgramDescription>
```

Select

Search and navigation server issues CRIDs or trailers of desired contents and sends them to the PDR on a bi-directional network.

Locate

PDR selects a CRID and sends the CRID to a location resolution server on IP network. The location resolution server resolves a locator or a schedule according to the CRID and sends it to the PDR.

Acquire

The PDR accesses a content server according to the locator or records the contents according to the schedule of the desired contents. The local storage management function will use any alternative locators to resolve recording conflicts. The chosen locator will then be used to tune to the specified channel at the specified time and record for the specified duration. To ensure that the content is recorded the system must monitor for changes in the location of the content.

View

The desired contents have been acquired they are made available for viewing. As the viewer may want to view the associated metadata at the time of playback, the system should store the associated metadata at the time of selection or capture.

Finishing

This may involve a user preference system storing information about the viewing of the contents. This information could then be used by an agent to modify the preferences of the user.

7.10 Example: Profile 1 bi-directional multi-provider ECG

This application shows the phase one *TV-Anytime* technologies for the combining of TVA metadata ECG sets from several metadata providers

A service provider delivers a range of channels to subscribers on a uni-directional satellite network. They provide to their subscribers by aggregating (from a variety of sources) a CRID, title, genre, synopsis, rating, and location resolution information in their uni-directional stream.

A third party is commercially contracted by the provider to supply (to the consumer via the return channel for an extra fee) an alternative and complete set of metadata for all feature films carried on that platform. The data set may duplicate information already available via the uni-directional service and fields may be identical to the previously delivered metadata but not necessarily so as the viewer is receiving added value (e.g. richer synopsis, rating/review, viewer ratings, cast lists etc.). The PDR user then selects and switches between either data set depending on their preference at that time. They activate capture for a particular film based on these data sets.

Publish

The service provider publishes CRIDs and basic programme information. By virtue of the contractual relationship of the third party to the service provider, the third party metadata uses the same CRIDs as used by the service provider.

Search

The PDR will issue a query for every CRID received via the uni-directional network that the PDR wishes to get more information on:

```
<get_Data>
 <QueryConstraints>
 <PredicateBag type='OR'>
  <BinaryPredicate fieldID='CRID' fieldValue='crid://broadcaster.com/1234' />
  <BinaryPredicate fieldID='CRID' fieldValue='crid://broadcaster.com/2345' />
  <BinaryPredicate fieldID='CRID' fieldValue='crid://broadcaster.com/2434' />
 …
```

```
 </PredicateBag>
 </QueryConstraints>
 <RequestedTables>
  <Table type='ProgramInformationTable'>
   <SortCriteria fieldID='tvaf:Title' />
  </Table>
  <Table type='CreditsInformationTable' />
  <Table type='ProgramReviewTable' />
 </RequestedTables>
</get_Data>
```

Select/Locate/View/Finish

As in previous examples.

7.11 Related material recording

The RelatedMaterialType DS can be used to link, for example, a promotional trailer to the content that it promotes. For example, a PDR may be viewing a piece of content and find a Related MaterialType DS in a data stream associated with the content. The encoding of the RelatedMaterialType into this data stream and the mechanism by which it is associated with the content is transport specific and outside the scope of TVA.

The RelatedMaterialType DS provides the crid of the material as well as a description of how this material relates to the trailer being viewed. In this example, "Trailer" indicates that this content is a small preview of the whole programme. Using this information the PDR may, in an implementation specific way, present the user with a choice to record this programme. Other types of HowRelated include, "The making of" and "Product purchase". Additionally, promotional text is included in the RelatedMaterial DS that the PDR may choose to display to provide the user with some context for why this programme may be chosen for recording.

Publish

Suppose a service provider publishes the following information as expressed by the XML snippet:

```
<?xml version="1.0" encoding="UTF-8"?>
<TVAContentLinks xmlns="urn:tva:metadata:2004" xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:tva:metadata:2004
.\tva_metadata_v13_am1.xsd">
   <RelatedMaterial>
      <HowRelated href="urn:tva:metadata:cs:HowRelatedCS:2004">
         <Name>Trailer</Name>
      </HowRelated>
      <MediaLocator>
         <mpeg7:MediaUri>crid://whatever</mpeg7:MediaUri>
      </MediaLocator>
      <PromotionalText xml:lang="en">Record Foxes episode 6</PromotionalText>
   </RelatedMaterial>
</TVAContentLinks>
```

Search/Select

When the promotional item, in this case a trailer for "Foxes episode 6" is presented, the user is given an option to record the associated programme using the CRID in the MediaLocator.

Note that this CRID can be used to retrieve further metadata, if so required.

Locate/Acquire/View/Finish

As per previous examples.

7.12 User Profiling Cookbook Scenario and Work-Through Application

A viewer buys a *TV-Anytime* compliant device for their home. When they use the device for the first time they are encouraged, being offered cheaper content and other services, to plug the device into the phone line. During the set-up process they are also asked to enter some basic information about themselves. This includes their family group (couple), roughly where they live (Provo) and their date of birth (August 1966). Also at this time they are asked whether they would like the TVA device to recommend content to them. They say yes and are taken to a simple set-up screen where they choose:

1) Favourite types of programmes (content) - News, Sport - athletics and gameshows.

2) Types of programmes (atmosphere) - alternative, breathtaking and inspirational.

3) A range of people they like (key talent). - Sean Connery, Humphrey Bogart, Clint Eastwood and Meg Ryan.

This information is moved into their user profile and some elements are made static and some can be updated.

The viewer then starts to use the device. At each point in their use of the device their profile is being updated.

a) At the start they request the capture of a drama "Pride and Prejudice" starring Colin Firth;

b) They also press "record" for a programme called "Blue Moon", a documentary about the Solar System narrated by Colin Firth; and

c) They then play and watch a local Provo News magazine programme which is transmitted with a second audio track in Spanish (which has been recorded for them already based on their initial favourites settings);

d) A Bond movie starring Sean Connery is discovered by the agent in the box and offers the programme as a suggestion.

During all of the above actions their user profile is being constantly updated.

This is an example of how a section of the Schema looks once the viewer has performed the functions above:

```
<TVAMain>
   <ClassificationSchemeTable>
      <CSAlias alias="ia"
         href="urn:tva:metadata:cs:IntendedAudienceCS:2004"/>
      <CSAlias alias="co"
         href="urn:tva:metadata:cs:ContentCS:2004"/>
      <CSAlias alias="or"
         href="urn:tva:metadata:cs:OriginationCS:2004"/>
      <CSAlias alias="in"
         href="urn:tva:metadata:cs:IntentionCS:2004"/>
      <CSAlias alias="at"
         href="urn:tva:metadata:cs:AtmosphereCS:2004"/>
      <CSAlias alias="ro"
         href="urn:tva:metadata:cs:RoleCS:2004"/>
   </ClassificationSchemeTable>
   <UserDescription>
      <UserPreferences>
         <mpeg7:FilteringAndSearchPreferences>
            <mpeg7:CreationPreferences>
               <mpeg7:Creator preferenceValue="100">
                  <mpeg7:Role href=":ro:ACTOR"/>
                  <mpeg7:Agent xsi:type="mpeg7:PersonType">
                     <mpeg7:Name>
                        <mpeg7:GivenName>Sean</mpeg7:GivenName>
                        <mpeg7:FamilyName>Connery</mpeg7:FamilyName>
                     </mpeg7:Name>
                  </mpeg7:Agent>
                  <mpeg7:Character>
                     <mpeg7:GivenName>James Bond</mpeg7:GivenName>
                  </mpeg7:Character>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="30">
```

```
                  <mpeg7:Role href=":ro:DIRECTOR"/>
                  <mpeg7:Agent xsi:type="mpeg7:PersonType">
                     <mpeg7:Name>
                        <mpeg7:GivenName>John</mpeg7:GivenName>
                        <mpeg7:FamilyName>Smith</mpeg7:FamilyName>
                     </mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="30">
                  <mpeg7:Role href=":ro:AUTHOR"/>
                  <mpeg7:Agent xsi:type="mpeg7:PersonType">
                     <mpeg7:Name>
                        <mpeg7:GivenName>Jane</mpeg7:GivenName>
                        <mpeg7:FamilyName>Austen</mpeg7:FamilyName>
                     </mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="50">
                  <mpeg7:Role href=":ro:BROADCASTER"/>
                  <mpeg7:Agent xsi:type="mpeg7:OrganizationType">
                     <mpeg7:Name>BBC</mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="30">
                  <mpeg7:Role href=":ro:BROADCASTER"/>
                  <mpeg7:Agent xsi:type="mpeg7:OrganizationType">
                     <mpeg7:Name>ITV</mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="30">
                  <mpeg7:Role href=":ro:BROADCASTER"/>
                  <mpeg7:Agent xsi:type="mpeg7:OrganizationType">
                     <mpeg7:Name>Sky</mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="60">
                  <mpeg7:Role href=":ro:ACTOR"/>
                  <mpeg7:Agent xsi:type="mpeg7:PersonType">
                     <mpeg7:Name>
                        <mpeg7:GivenName>Humphrey</mpeg7:GivenName>
                        <mpeg7:FamilyName>Bogart</mpeg7:FamilyName>
                     </mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="60">
                  <mpeg7:Role href=":ro:ACTOR"/>
                  <mpeg7:Agent xsi:type="mpeg7:PersonType">
                     <mpeg7:Name>
                        <mpeg7:GivenName>Meg</mpeg7:GivenName>
                        <mpeg7:FamilyName>Ryan</mpeg7:FamilyName>
                     </mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="60">
                  <mpeg7:Role href=":ro:ACTOR"/>
                  <mpeg7:Agent xsi:type="mpeg7:PersonType">
                     <mpeg7:Name>
                        <mpeg7:GivenName>Clint</mpeg7:GivenName>
                        <mpeg7:FamilyName>Eastwood</mpeg7:FamilyName>
                     </mpeg7:Name>
                  </mpeg7:Agent>
               </mpeg7:Creator>
               <mpeg7:Creator preferenceValue="60">
                  <mpeg7:Role href=":ro:ACTOR"/>
                  <mpeg7:Agent xsi:type="mpeg7:PersonType">
                     <mpeg7:Name>
                        <mpeg7:GivenName>Peter</mpeg7:GivenName>
                        <mpeg7:FamilyName>Firth</mpeg7:FamilyName>
                     </mpeg7:Name>
```

```
               </mpeg7:Agent>
            </mpeg7:Creator>
         </mpeg7:CreationPreferences>
         <mpeg7:ClassificationPreferences>
            <mpeg7:Genre preferenceValue="20" href=":ia:4.2.2.2">
               <mpeg7:Name>Age 25-34</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="20" href=":co:3.1.1">
               <mpeg7:Name>News</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="80" href=":co:3.2.1">
               <mpeg7:Name>Sports - Athletics</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="80" href=":co:3.5.1">
               <mpeg7:Name>Amusement- Gameshows</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="80" href=":or:5.1">
               <mpeg7:Name>Studio</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="20" href=":in:1.1">
               <mpeg7:Name>Entertainment</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="40" href=":at:8.1">
               <mpeg7:Name>Alternative</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="40" href=":at:8.6">
               <mpeg7:Name>Breathtaking</mpeg7:Name>
            </mpeg7:Genre>
            <mpeg7:Genre preferenceValue="40" href=":at:8.30">
               <mpeg7:Name>Inspirational</mpeg7:Name>
            </mpeg7:Genre>
         </mpeg7:ClassificationPreferences>
      </mpeg7:FilteringAndSearchPreferences>
   </UserPreferences>
</UserDescription>
</TVAMain>
```

The following is a table of the TVAF recommended user profile data sets for each of the programmes.

Title	Pride and Prejudice	Blue Moon	News at Ten	Goldfinger
CONTENT	3.4.14 Fiction, Period Drama	3.1.6.5 Non Fiction, Sciences, Space/Universe	3.1.1 Non Fiction, News	3.4.6.1 Fiction, Action, Adventure
ORIGINATION	5.2.3 Made on Location, Edited	5.2.3 Made on Location, Edited	5.1.1 Studio, Live	5.3 Cinema Originated
INTENTION	1.1 Entertainment	1.8 Enrichment	1.2 Information	1.1 Entertainment
ATMOSPHERE	8.24 Heart Rending, 8.25 Heartwarming 8.39 Romantic 8.41 Sad, 8.48 Stunning	8.30 Inspirational, 8.15 Edifying, 8.4 Ambitious, 8.2 Analytical 8.48 Stunning	Null field	8.20 Gripping, 8.17 Fast moving, 8.22 Gutsy, 8.38 Rollercoaster 8.44 Sexy, 8.45 Stunning 8.51 Thriller
INTENDED AUDIENCE	Age Group 4.2.2 Adults, Gender 4.63, both MandF, Geographical 4.75 Universal, Language English (en)	Age Group 4.2.3 All Age Groups, Social Group 4.4.1 AB and 4.4.2 C1C2, Gender 4.6.3 Both MandF, Geographical 4.71 Universal, Language English (en)	Age Group 4.2.2 Adults. Geographical 4.7.5 Local - (Keyword Sunnyvale) Language English (en) and Spanish (es)	Age Group 4.2.1.3 Age 14-15 and 4.2.2 Adults, Gender 4.6.1 Male, Geographical 4.7.1 Universal, Language English (en)
BROADCASTER (from MPEG 7 RoleCS)	BBC	NOS	ITV	Sky
DIRECTOR (From MPEG7 RoleCS)	John Smith	Null field	Null field	Cubbi Broccoli
KEY TALENT (From TVA RoleCS)	Peter Firth	Peter Firth	*Null Field*	Sean Connery
KEY CHARACTER (From TVA RoleCS)	Null Field	Null Field	*Null Field*	James Bond
Author From (MPEG7 RoleCS) (WRITER)	Jane Austin	*Null field*	*Null field*	Ian Fleming

When inbound content data enters the box, an agent in the box might use the previously captured data from viewed programmes to match. Using the above examples of data, the agent might assume that as two programmes had Peter Firth as a Key Talent the viewer would like to watch other programmes in which he takes part. If Atmosphere Stunning appears in three out of four programmes and would be high on the list to look out for. In the weighting environment the agent might automatically assign Stunning the maximum positive value.

Data to be returned to the service provider or Broadcaster

Providers of content wish to know how their content is performing as far as the consumer is concerned. Business decisions will be made as a result (e.g. cancel the scheduling of a programme because the viewers do not like it; not enough people are watching a particular strand; the viewer really likes astronomy documentaries but they are not providing any, advertisers will want to know viewers numbers, appreciation index across demographics etc.).

The following information might be useful:

1) Actual viewing **of their own** programmes by receiving a list of locators/CRIDs that refers to their own programmes only.

2) An anonymous aggregated user profile consisting of a sub set of fields that will indicate the viewing habits of the viewing population across **all** content.

The first one is out of scope for Phase 1 because no authenticated return channel RMP has been specified:

1) A subset of the data above could be returned to the broadcaster if this functionality has been enabled by the consumer or required by the consumer's contract with the service provider. A minimum useful set of data might include the following elements:

   Content, Origination, Intention, Atmosphere and Intended Audience

Each with the associated aggregated rating. This will enable service providers and broadcasters the ability to analyze aggregated viewer preferences and allows them to make business decisions based on their performance in these areas.

Issues surrounding rating of data.

MPEG7 define the rating scale between a minimum of –100 and a maximum of +100 with a no preference value set at 0.

To make systems interoperable and viewer preferences portable (e.g. when they buy a newer box or wish to take their profile to a remote location - on holiday, second home etc.) it is important that systems comply with this scale, i.e. there must be both positive and negative values which are 100 even if only a limited number of choices between these values are implemented.

EXAMPLE:

-100 -80 -60 -40 -20 -0 20 40 60 80 100

or

-100 -66 -33 0 33 66 100

or

-100 50 0 50 100

in the implementation in the box these can be **labelled** as implementers' wishes… e.g.:

0 1 2 3 4

or

really do not like do not like neutral quite like like a lot

8 Issues per phase

This clause gives some explicit implementation hints for less obvious issues in the different *TV-Anytime* phases. Not all phases are covered in the present document, since it focuses at the use of content referencing.

8.1 General System issues

To get an actual *TV-Anytime* system into the phases described in the previous clauses some generic system issues need to be solved. These issues are out of the scope of *TV-Anytime*, but some of them are presented here to ease the implementer's life. Most issues are related to actual PDR implementations. The issues fit into a number of categories discussed below.

8.1.1 Set-up and service discovery

In the set-up phase of a PDR, e.g. when the user first installs a box in his home, a number of things need to be done to allow for it to find the *TV-Anytime* metadata and location resolution services. The way that this works is transport dependent and no complete descriptions for all transports are given, the analogue case and the digital DVB satellite case are discussed below.

Analogue environment

For content resolution to work in an analogue environment a mapping between a content reference ID and a physical channel frequency, date and time or duration needs to be made. This information can be carried via IP in the VBI, for example, or via a smart card or floppy disc. In the VBI case the box only needs a well know port number to do service discovery. After the location of the resolution tables and resolution authority records are known, download of those can commence. Metadata is probably too bulky to be carried in the VBI, so other means of getting that into the box need to be there, e.g. a telephone line.

For the analogue situation a preferred way of issuing locators would be, for example:

analog:cablenetworkXYZ/broadcaster/8am/1hour.

This would allow the broadcaster to use the same locators across multiple cable networks since frequency allocations will be different. For this to work the box needs to be able to make the mapping between channel name and physical channel frequency. This can be done during set-up by the user or, for example, in a Western-European PAL environment by looking at VBI information carrying Programme Delivery Control for analogue VCRs. The same trick would also be useful in a digital environment with broadcasters being on a number of different media like satellite, cable etc.

Similarly for metadata, the box needs to be able to find out where metadata is carried. This could be done via the VBI, e.g. using teletext or PDC.

Digital DVB environment

In the DVB environment, operation is similar but for the actual locators that are put into the location resolution tables. In the DVB environment service information, program IDs, event triggers and start-stop times could, for example, be used. To get the location resolution authority records, the box needs to know where to look in the transport stream.

One solution could be that there is a well-known service, e.g. "*TV-Anytime* services" in the transport containing pointers to all data needed by the box, e.g. it would point to the location of location resolution authority records and actual location resolution tables.

Another possibility is that each broadcast service contains its own "*TV-Anytime* services" entry pointing at the relevant tables.

In the DVB environment containers should be reserved, e.g. tables or sections that could carry the pointer to the TVA metadata. The current TVA specifications do not cater for this.

8.1.2 Transport and delivery of *TV-Anytime* data

The transport and delivery of TVA data requires the definition and specification of a set of technical features currently not covered by the *TV-Anytime* specification. These features can be summarized as follows:

- *Identification and signalling*: an identification mechanism (e.g. a metadata location descriptor corresponding to a specific metadata format) is needed to associate specific resources to the transport of TVA data, and to signal that data being transported is, for example, TVA data. Signalling can also be used to inform the system about the presence of incoming TVA data.

- *Location*: a mechanism (e.g. a resolving authority record or a metadata locator) that allows to point at the actual location of the data container from which location resolution data or metadata will be retrieved. The definition of a locator needs to take into account the transitory nature of the associated information.

It should allow identifying, signalling and locating TVA data carried over a variety of transport systems (e.g. MPEG TS - over the PES, a data or object carousel, in metadata section, or over IP).

8.1.3 Updating resolution tables

Both the Resolving Authority Record (RAR) tables and the Resolution tables need to be kept up to date. The Resolving Authority Records contain information about the various resolution services available to the PDR device. The Resolution tables contain the mappings of CRIDs to other CRIDs or to locators. In the broadcast scenario, these tables must be pushed to the box, most likely using a broadcast carousel. Whole tables should be sent regularly to provide for new PDRs entering the system, but incremental updates are also useful, as they save bandwidth and also (probably) require less processing power at the client device when they are received.

Changes in the version field of the RAR (made by the location resolution provider) indicate to the client an update of all RARs for a given authority serviced by this provider. The expiry date of a RAR is an additional trigger for updates. The updating of location resolution tables is tied to the transport mechanism.

Below is an example of updating RARs where a resolution provider has several resolution services available in the same box. In the example, the broadcaster HBC has three channels on the multiplex (HBC1, HBC2 and HBC Gold, which are available on dvb://1.2eef.3f5, dvb://1.2eef.106 and dvb://1.2eef.3f5, respectively). A fourth channel is provided by another broadcaster who is also a resolution provider (broadcaster.co.jp). This channel is available at dvb://1.104.e5f.

The PDR has four RARs stored, three of which point to resolution services provided by HBC, and the other RAR that points to the resolution service of the other broadcaster.

	RAR A	**RAR B**	**RAR C**	**RAR D**
RAR fields				
Authority	hbc.com	creator.com	creator.com	creator.com
Resolution Provider	hbc.com	hbc.com	hbc.com	broacaster.co.jp
URL	dvb://1.2eef.3f5	dvb://1.2eef.106	dvb://1.2eef.3f5	dvb://1.104.e5f
Version	24	2	2	96

RAR A refers to CRIDs created by the broadcaster hbc.com who is also a resolution provider for hbc.com CRIDs. RAR B, RAR C and RAR D all refer to resolving CRIDs created by the creator.com CRID authority.

If the PDR receives a RAR which contains:

Authority	creator.com
Resolution Provider	hbc.com
URL	dvb://1.2eef.3f6
Version	3

the PDR will discard all the RARs that have authority equal to creator.com and provider equal to hbc.com. In this example, the RAR above would cause RAR B and RAR C to be discarded and the new RAR to be stored in the PDR.

	RAR A	**RAR D**	**RAR E**	
RAR fields				
Authority	hbc.com	creator.com	creator.com	
Resolution Provider	hbc.com	broacaster.co.jp	hbc.com	
URL	dvb://1.2eef.3f5	dvb://1.104.e5f	dvb://1.2eef.3f6	
Version	24	96	3	

If the resolution provider wanted to update RAR B and keep RAR C as it is, they will need to transmit new versions of both RARs.

	RAR A	**RAR D**	**RAR E**	**RAR F**
RAR fields				
Authority	hbc.com	creator.com	creator.com	creator.com
Resolution Provider	hbc.com	broacaster.co.jp	hbc.com	hbc.com
URL	dvb://1.2eef.3f5	dvb://1.104.e5f	dvb://1.2eef.3f6	dvb://1.2eef.3f5
Version	24	96	3	3

8.1.4 Updating metadata

Currently there are no mechanisms defined that allow for update of part or all of the metadata.

8.2 Publishing phase

Re-run/repeat of content

A re-run is defined as content that is broadcast at different times to suit user convenience, sometime subsequent to its original broadcast. A repeat is generally regarded as old content. The content service provider may package content being re-shown within a short period of the original broadcast as a repeat, and leave the associated metadata untouched. If it is some months or years later, the content service provider may package it as a re-run and alter the associated metadata. The consumer may regard it as original viewing, as a re-run or as a repeat of the same content. Regarding the creation and publication of a CRID to reference the re-run/repeat, a number of scenarios can be envisaged:

1) The original CRID may be re-used, but with a new locator. Also, alternative space and time locations may be provided for the content after resolution. The PDR can take advantage of multiple location options to resolve recording conflicts. The use of the same CRID, if the content service provider always uses unique CRIDs, is also a way for the box to identify the item as having been previously consumed. If there is additional metadata available for this particular airing, instance metadata can be used.

2) The content service provider may issue a new CRID to refer to the re-run/repeat. From the PDRs perspective at least, the content is then regarded as different. The content service provider may package the programme with new metadata, for example, if a motion picture actor dies, then her films may be re-run, the fact that she has died being added to the metadata surrounding the movie.

3) A third party may issue a group CRID to refer to all airings of an item across many different service providers. This has the advantage that the local storage management can use the group CRID to help solve recording conflicts. From the consumer's perspective, each occurrence is effectively a repeat even though they occur across service providers.

4) Instance Metadata Identifier.

For the case when a broadcast is to be repeated (or re-run) using the same CRID, the metadata provider may wish to differentiate between these publication instances using the instance description metadata. This differentiation can be communicated to the viewer through EPG-like rendering on screen or utilized by a software agent acting on the viewer's behalf.

When allocating Instance Metadata Identifiers to the instance descriptions, the metadata providers can choose to use an Instance Metadata identifier from its own proprietary namespace. As such the <name> field of the Instance Metadata Identifier must be included and arrangements made to supply these Instance Metadata Identifiers to collaborating resolution providers.

Alternatively, the Instance Metadata Identifiers may be supplied by the CRID authority and used within the metadata without the <name> field of the Instance Metadata Identifier syntax. Resolution providers may also have access to these instance metadata identifiers from the CRID authority and are able to supply the corresponding Instance metadata identifiers for each of the locators within the CRID location resolution table.

Note that it is always possible for a location resolution provider to provide CRID location resolution tables for the CRID without any Instance Metadata Identifiers.

8.3 Search and select

Metadata from multiple providers

It is envisaged that multiple metadata providers will / can provide metadata for the same CRID. This information then could be linked via the CRID. However, there needs to be a metadata provider field, which is currently absent from the metadata specification.

Selection on basis of time/channel

"What's on at 8 o'clock tonight?"

The scenario depends on a few implementation issues (bullets 1 and 2) and/or how service providers will provide required metadata (bullets 3 and 4):

1) With a location resolution service that, when given a CRID, only provides locators, the box would have to resolve all CRIDs and search for all 8 o'clock entries.

2) If the box has access to the stored location resolution tables this is a straightforward query.

3) The service provider could issue a (group) CRID containing all the contents available at a given time. This is a limited solution, similar to a restricted EPG.

4) If the timing information is sent separately in the metadata stream (e.g. using instance metadata), there may be conflicts between the metadata and the location resolution data.

Note that content that is already recorded is available at any time.

Search and select based on metadata like cost

"I want to see a free version of the film "Lizy, Queen of the Desert".

In a *TV-Anytime* system there will be a means of conveying associated costs with a certain piece of content. A user can use that information to choose a piece of content that is e.g. free, or has a lower price due to the scheduling. For example, a boxing match could be cheaper as a repeat than live.

One solution to this problem is to create distinct CRIDs for each showing of the boxing match.

There will be different CRIDs for the same programme with the same metadata but with different cost attached. The capture of one CRID will lead directly to the required location with associated cost.

In that case the instance metadata identifier can be used. The Instance metadata identifier is the link between an actual location and the metadata associated with that location. A service provider could assign an instance metadata identifier to a certain location that has a certain price. After this the content resolution process could match up the CRID - instance metadata identifier combination, so that the precise instance of the programme can be captured, even if it changes location. This has the advantage that metadata need not be duplicated across multiple instances of the same content.

Essentially the same content has different CRIDs

"I want to get the "Foxes" comedy show".

The same content, at least in the eyes of the consumer, can have different CRIDs, because different service providers might decide to do so. Three options arise to deal with this using metadata, third parties or unique identification, respectively:

1) A metadata search will return with multiple choices, with different CRIDs. The choice between them would depend on the available metadata. For example, the title field in the ProgramInformationTable could be used to conduct such a search. Without a connection between the CRIDs of the content items grouping would have to be done in the box.

2) A metadata aggregator could generate a (group) CRID, which refers to all of the different CRIDs. This creates a single point reference for the content.

3) If the same content could be identified uniquely, the different offerings may be tied together in the box. Such a unique identifier could be carried in the OtherIdentifier field in the ProgramInformationTable.

How to find a specific episode

"I want to see episode 15 of the original "Foxes" series".

1) This is straightforward, if the index attribute in the programme description data contains the episode number.

2) The programme description data (e.g. title or synopsis) could contain the appropriate descriptive text for a textual search. Inconsistencies in phrases used may limit the use of a textual search.

3) If the (group) CRID returns an ordered list of CRIDs one could infer the episode number. This is a limited solution and is not currently part of the Content Referencing specification.

How to identify the latest episode of Foxes

"I want the latest episode of "Foxes".

This scenario can be implemented in the following ways:

1) A service provider generates a CRID, which points to the latest episode within a service.

2) A third-party generates a CRID, which points to the latest episode on all services.

3) Choosing a (group) CRID will lead to the capture of the next available episode.

Have I seen this content before?

"I do not want to record this if I have seen it before".

A box may store what the user has seen. If the box stores the CRID of the programme and all CRIDs were unique for all time, this would be sufficient. However, different service providers will use different CRIDs for the same content. Also service providers may issue a new CRID if the content has been changed only slightly or they want to promote the programme in a particular way. The system may need to store programme description data in order to fulfil this requirement. Alternatively, if a unique identifier is available in the OtherIdentifier field, this could be used for this purpose.

8.4 Location phase

"Make locator names unambiguous"

For example two satellites feed one box. In this scenario how does the system distinguish between say channel 5 from each satellite?

Or, phrased differently, what happens when the same service is on both physical inputs?

1) Box implementation issue. If the content on both feeds has the same CRIDs, the box can decide to listen to whichever feed based on some criteria (e.g. random "flip-a-coin criteria") or the PublicationType element in the user preferences DS can be used.

2) If CRIDs are different between the two feeds, which implies the content to be different (differences in quality, encoding type,). The user will be the one deciding which one to listen to.

"Re-resolution"

Currently there is no best practice defined for re-resolving locations. The only way to make sure that a PDR does not miss scheduling changes is to check back with the resolution tables every time they are updated. This drives the requirements for carriage of the data, e.g. frequency of re-transmission of location resolution data in the broadcast environment. Proper practice is to monitor the location resolution data, not the instance metadata.

"Locate instance at a specific location"

Using the instance metadata identifier it is possible to select an instance at a specific location for capture. To determine the correct locator, the CRID together with the instance metadata identifier should be used.

8.5 Acquire phase

Acquisition of metadata with content and/or separate from content

Some metadata may be related to the actual timeline of the content, e.g. metadata that needs to show up at a certain point in the programme. Currently there is no way of synchronizing (Meta) data with content in the *TV-Anytime* context.

Validation of content

On validation of the acquired content the following points are identified:

- Trustworthiness of the resolving authority may be assumed.
- It is impossible to attach the CRID used for resolving the content in all cases.
- Other means of identification may be needed (e.g. V-ISAN, ISAN, Broadcaster own ID, …).
- However, it also is impossible to attach a globally registered ID in all cases.

Validation is possible if all "leaf" CRIDs are attached to the content, which is easily achievable when only working in an environment involving a single service provider.

Rights management

Rights management topics such as Access Control, Content and Copy Protection need to be addressed in the acquisition phase once the Rights Management specification is developed.

Programme Delivery Control, signalling resolution updates

The resolution engine may inform the recording management unit of the latest updates in the delivery timing of the content (see clause 8.1.3). Alternatively the recording manager may poll the resolution engine. Recording management is an implementation issue and hence beyond the scope of the present document. However, service providers and box implementers are required to provide an accurate mechanism to update changes in the schedules similar to PDC.

It is suggested that more accurate timing may be achieved by using "triggers" or other equivalent mechanisms in the transport stream e.g. DVB event ID. "Triggers" can be part of locator syntax. It is noted that Triggers will be transport dependent, e.g. DVB, ATSC, and ARIB.

Annex A (informative): Example of Usage History DS

The following example highlights the usage history for the "John Smith" user. During the observation period two episodes of the "Fox" series were recorded and subsequently viewed. During the viewing of the "Red Foxes" episode the user zoomed in twice. Finally the user previewed the "Blue Foxes" episode.

```
<UsageHistory id="usage-history-001" allowCollection="true">
   <mpeg7:UserIdentifier protected="true">
      <mpeg7:Name xml:lang="en">John Smith</mpeg7:Name>
   </mpeg7:UserIdentifier>
   <mpeg7:UserActionHistory id="useraction-history-001" protected="false">
      <mpeg7:ObservationPeriod>
         <mpeg7:TimePoint>2001-02-02T18:00-08:00</mpeg7:TimePoint>
         <mpeg7:Duration>PT96H</mpeg7:Duration>
      </mpeg7:ObservationPeriod>
      <mpeg7:ObservationPeriod>
         <mpeg7:TimePoint>2001-02-02T18:00-08:00</mpeg7:TimePoint>
         <mpeg7:Duration>PT6H</mpeg7:Duration>
      </mpeg7:ObservationPeriod>
      <mpeg7:UserActionList id="ua-list-001" numOfInstances="2" totalDuration="PT2H30M">
         <mpeg7:ActionType
         href="urn:tva:metadata:cs:ActionTypeCS:2004:1.2">
            <mpeg7:Name>Record</mpeg7:Name>
         </mpeg7:ActionType>
         <mpeg7:UserAction>
            <mpeg7:ActionTime>
                <mpeg7:MediaTime>
                   <mpeg7:MediaTimePoint>2001-02-02T19:00:00</mpeg7:MediaTimePoint>
                   <mpeg7:MediaDuration>PT1H</mpeg7:MediaDuration>
                </mpeg7:MediaTime>
            </mpeg7:ActionTime>
            <mpeg7:ProgramIdentifier organization="TVAF" type="CRID">
                crid://broadcaster.com/RedFoxesCrid ForThisEpisode>
            </mpeg7:ProgramIdentifier>
         </mpeg7:UserAction>
         <mpeg7:UserAction>
            <mpeg7:ActionTime>
                <mpeg7:MediaTime>
                    <mpeg7:MediaTimePoint>2001-02-03T19:00:00</mpeg7:MediaTimePoint>
                    <mpeg7:MediaDuration>PT1H</mpeg7:MediaDuration>
                </mpeg7:MediaTime>
            </mpeg7:ActionTime>
            <mpeg7:ProgramIdentifier organization="TVAF" type="CRID">
                crid://broadcaster.com/GreyFoxesCrid ForThisEpisode
            </mpeg7:ProgramIdentifier>
         </mpeg7:UserAction>
      </mpeg7:UserActionList>
      <mpeg7:UserActionList id="ua-list-002" numOfInstances="25" totalDuration="PT7H02M">
         <mpeg7:ActionType
         href="urn:tva:metadata:cs:ActionTypeCS:2004:1.2">
             <mpeg7:Name>View</mpeg7:Name>
         </mpeg7:ActionType>
         <mpeg7:UserAction>
             <mpeg7:ProgramIdentifier organization="TVAF" type="CRID">
                 crid://broadcaster.com/RedFoxesCrid ForThisEpisode
             </mpeg7:ProgramIdentifier>
         </mpeg7:UserAction>
         <mpeg7:UserAction>
             <mpeg7:ActionTime>
                 <mpeg7:MediaTime>
                     <mpeg7:MediaTimePoint>2001-02-04T20:30:00</mpeg7:MediaTimePoint>
                     <mpeg7:MediaDuration>PT1M45S</mpeg7:MediaDuration>
                 </mpeg7:MediaTime>
             </mpeg7:ActionTime>
             <mpeg7:ProgramIdentifier organization="TVAF" type="CRID">
```

```
                crid://broadcaster.com/GreyFoxesCrid ForThisEpisode
            </mpeg7:ProgramIdentifier>
        </mpeg7:UserAction>
    </mpeg7:UserActionList>
    <mpeg7:UserActionList id="ual-003" numOfInstances="2" totalDuration="PT10S">
        <mpeg7:ActionType
        href="urn:tva:metadata:cs:ActionTypeCS:2004:1.2">
            <mpeg7:Name>Zoom</mpeg7:Name>
        </mpeg7:ActionType>
    <mpeg7:UserAction>
    <mpeg7:ProgramIdentifier organization="TVAF" type="CRID">
        crid://broadcaster.com/RedFoxesCrid ForThisEpisode </mpeg7:ProgramIdentifier>
    </mpeg7:UserAction>
    </mpeg7:UserActionList>
    <mpeg7:UserActionList id="ual-004" numOfInstances="1">
        <mpeg7:ActionType>
            <mpeg7:Name>Preview</mpeg7:Name>
        </mpeg7:ActionType>
        <mpeg7:UserAction>
            <mpeg7:ProgramIdentifier organization="TVAF" type="CRID">
                crid://broadcaster.com/BlueFoxesCrid ForThisEpisode
            </mpeg7:ProgramIdentifier>
        </mpeg7:UserAction>
    </mpeg7:UserActionList>
  </mpeg7:UserActionHistory>
</UsageHistory>
```

Annex B (normative): Transport environment and requirements

This annex is a mandatory part of the *TV-Anytime* set of specifications.

B.1 Scope

The current *TV-Anytime* specifications describe the information structures of content referencing and metadata and how these two work together in a system context. There are however a lot of features needed to deliver *TV-Anytime* services that are not specified in the *TV-Anytime* environment. Examples of these features are the actual carriage of the *TV-Anytime* data, exact start and stop times of programmes, the carriage of the content, the linkage of *TV-Anytime* metadata timelines to actual transmission timelines etc.

The goal of the transport interface specification is define the requirements that *TV-Anytime* services need from the lower layers. Since there may be many lower layer technologies used in the deployment of *TV-Anytime*, *TV-Anytime* relies on others to actually implement those requirements. *TV-Anytime* will define a transport agnostic way to get its data across, which can be used in different networks and regions in the world.

The first transport requirements specification focus on uni-directional access, e.g. broadcast networks or unicast or multicast over IP networks. So it will consider requirements for networks without a back channel, services requiring bi-directional connectivity will be considered in a later phase.

B.2 Requirements on the Uni-Directional Delivery System

The underlying delivery system shall:

1) Enable the transport of valid *TV-Anytime* information asynchronously to programmes and potentially split across different transport mechanisms. I.e. not all *TV-Anytime* data needs to be encapsulated in the same manner.

2) Provide a method to locate where and what type of *TV-Anytime* information is being carried, i.e.:

   a) To acquire content resolution information, the location of the RAR - TS 102 822-4 [3] - is the required information from the transport layer.

   b) To acquire metadata, the information required from the transport layer are:

      - the list of available TVA metadata fragment streams;
      - the signalling of any modification occurring on them such as the addition of a new TVA metadata fragment stream or the removal of an existing one;
      - the fragment types or categories of TVA fragments carried in each TVA metadata fragment stream and information about their respective scope (e.g. list of broadcast channels, specific CRID);
      - the location of their respective entry points, namely the "TVAInit" message and possibly the "TVAMain" fragment if delivered separately;

- For the containers of each metadata fragment streams, the transport layer shall:
  - signal the Id of each container, its type and its location;
  - identify the version of each container - The current version of each container shall be signalled, and this shall be incremented whenever the contents of a container change;
  - allow monitor at a single point for version changes to a container. Ideally it should be possible to monitor just data containers, or if provided, just containers forming a single index;
  - allow to download all TVA metadata description containers (preferably in a parallel manner).

3) Provide locators (as specified in TS 102 822-4 [3]) for identification and location of content instances. Locators are also required to access fragments of *TV-Anytime* information.

4) Enable the insertion of certain types of *TV-Anytime* data (content referencing CRIDs, RMP information and metadata) along with the audiovisual content. *TV-Anytime* will define the syntax and semantics of such data.

5) Provide a mechanism to map from the linear timeline used by the *TV-Anytime* segmentation information to positions in the actual content (e.g. NPT). A linear timeline is used on a captured piece of content for segmentation purposes to enable random access to segments of the content.

6) Provide a signalling mechanism to accurately capture a piece of content referenced by a CRID, even if this content is interleaved with other pieces of content.

7) Enable the repeated transmission of *TV-Anytime* data. Repetition rates shall be flexible and it shall be possible to vary repetition rates for different types of *TV-Anytime* information. It shall not be necessary to wait for a whole repetition period of the whole *TV-Anytime* dataset to start decoding *TV-Anytime* data.

8) Support the selective updating of *TV-Anytime* information. The underlying delivery system shall not limit the *TV-Anytime* data size and allow for flexible update unit size for different *TV-Anytime* data types. Updates shall not cause data inconsistency at the receiver, even if previous updates have been missed. For efficient operation the underlying delivery system should provide an easy means of signalling updates of *TV-Anytime* information.

9) Accommodate for potential limited processing capability and memory at the receiver. It shall encapsulate *TV-Anytime* data in such a way that graceful recovery from transport errors is possible. The underlying delivery system may need to provide "wall clock time" to enable comparison of usage data.

10) Provide a means for transporting *TV-Anytime* information that is robust to transmission errors, so that the *TV-Anytime* data is received error free or is signalled to be in error.

Annex C (informative): Bibliography

Documents are available from the *TV-Anytime* web site http://www.tv-anytime.org.

List of figures

List of tables

History

Document history		
V1.1.1	October 2003	Publication
V1.2.1	September 2004	Publication