IST-1999-11702:

myTV

DELIVERABLE #4

Progress in Standardization

- Security level : Public -

Version	Status	Date	Comments
0.1	1st Draft	25 August 2000	For project partner comment and review
0.2	2nd Draft	5 September 2000	Annexes attached, text made consistent
0.3	3rd Draft	26 September 2000	Comments from Peter Mulder and Chris Newell included, workplan and AN110 attached, some more clean-up.
0.4	FINAL	30 September 2000	Minor editorial changes. Deliverable approved by all project partners

Project Number	:	IST-1999-11702
Project Title	:	myTV
Deliverable Type	:	Public

CEC Deliverable Number	:	
Internal Project Number	:	mytv-P-0039-1
Contractual Deliverable Date	:	30th September 2000
Actual Date of Deliverable	:	
Title of Deliverable	:	Progress in Standardization
Contributing Workpackages	:	wp100,200,300,400
Nature of Deliverable	:	report
Editor	:	Ronald Tol (ronald.tol@philips.com)

Abstract:

This report describes the progress the myTV project made in the two main standardization bodies relevant to the project: the TV-Anytime Forum and DVB. It discusses the main achievements in roughly the first year of the project. It further shows the work to be done during the remainder of the project. The actual contributions of the project are attached as annexes.

Keyword List:

TV-Anytime, TV, DVB, MHP, Storage.

Table of Contents

1 Introduction

The myTV project set out to

- Implement TV-Anytime for DVB, and continue to contribute to its standardisation
- Build different myTV boxes and TV-Anytime navigators, based on an agreed common architecture in order to demonstrate interoperability.
- Implement various new application concepts relying on TV-Anytime, all intended to facilitate personalisation of the TV offer for consumers.

The next sections cover the progress made in the two standardization bodies that are most important to the project: the TV-Anytime Forum and DVB. The appendices contain the myTV contributions proposed to the TV-Anytime Forum.

The remainder of this section briefly describes which myTV project members contributed to which standardization effort.

Heikki Pentikainen (Nokia), Peter Mulder (NOB) and Frans de Jong (NOB) participate in the metadata group meetings regularly. The latter two have also contributed various documents on behalf of NOB. Peter is also one of the metadata taskforce members in the SMPTE metadata resolution taskforce and Co-Chair of the MPEG-7 Metadata integration AdHoc Group.

David Wilson (NDS) is the head TVAF document editor for both the requirements series and the specifications series.

Cathy Toscan (Nokia Home Communications) participated in the TV-Anytime Forum Business Models work group at the 7th TVA meeting in Geneva, Switzerland in July 2000 and the 8th TVA meeting in Los Angeles, California in September 2000.

Alex Ashley (Philips Research Redhill) and Nigel Earnshaw (BBC) participate in the TV-Anytime Content Referencing workgroup. Alex is also the editor of the Content Referencing specification.

Ronald Tol (Philips Research Eindhoven) and Andrew McParland (BBC) participate in the System Design group. Ronald is also the editor for the System Design specification

Both Philips and NDS are represented in the Rights Management and Protection working group.

Nokia is represented in the DVB TAM group. Kimmo Löytänä has been participating in the DVB TAM group and its subgroups.

The BBC is represented in both the DVB Commercial Module MHP group and the Technical Module TAM group. Chris Newell of BBC R&D participates in the work of the TAM group and has been the editor of the "Evaluation of the MHP Specification against the Commercial Requirements" and the "Technical requirements for the Internet Access profile". The BBC has also contributed "Requirements for Home Recording and TV-Anytime" to the MHP group.

Philips is represented in the DVB TAM group and in the DVB CM. Jon Piesing is chair of the Java sub-group of TAM in DVB were the Java APIs for the DVB Multimedia Home Platform are defined.

Philips started a process to create an opening to the TM of DVB to standardize results of TV-Anytime and the myTV project.

2 TV-Anytime Forum

The TV-Anytime Forum is an association of organisations which seeks to develop specifications to enable audio-visual and other services based on mass-market high volume digital storage. The association comprises member organisations from Europe, the USA and Asia. Membership is open to all who sign the Memorandum of Understanding and attend meetings.

The Forum was formed at an inaugural meeting in Newport Beach, California from the 27 - 29 September 1999 and is now working to develop open specifications designed to allow Consumer Electronics Manufacturers, Content Creators, Telcos, Broadcasters and Service Providers to exploit high volume digital storage in consumer platforms.

At the time of writing over 120 companies signed the Memorandum of Understanding.

From September 1999 onwards bi-monthly meetings were held to prepare a Call for Contributions, asking for contributions on content referencing, metadata, rights management, business and system models. Subsequently, the forum started working on requirements documents in the different work areas incorporating the contributions resulting from the CfC. Currently, the forum works on specification documents. The workplan is given in Chapter 5.

2.1 Business models

The Business Models group of the TV-Anytime Forum is the only work group with a non-technical base. The requirements document which has been produced by Business Models (R1) is of informative nature. On the other hand, the specifications document currently being created by that group (S-1) will be a normative report, which the technical work groups will be asked to implement. The Business Models work group is headed by Gary Hayes from the BBC with help from Skip Pizzi from Microsoft, who has been the principle editor of the documents.

R-1 The TV-Anytime Environment (document TV035r6)

"The document attempts to define the business environment and usage scenarios in which personal digital media storage will exist in the near future. Because the personal digital recorder (PDR) is widely expected to become an extremely popular device, TV-Anytime believes it is critical to begin this analysis from the consumer's perspective. Therefore this document conducts its analysis of the market space for the PDR in a highly consumer-centric manner.

Although this is not a 'requirements document' in the strictest sense, it does present the attributes of an idealized personal media storage system from the consumer's point of view. It thereby provides the strongest and most appropriate foundation for such an environment, and is the basis for defining 'true' requirements for the milieu of the PDR, and understanding how it can be optimized by the TV-Anytime standardization process." [1]

Since this document is the first in the requirements series, it has also assumed a role as an introduction to the TV-Anytime Forum and the process that it has launched with it's Call for Contributions in December 1999. It includes sections that deal with:

1. opportunities and benefits for consumers, content owners, service providers, advertisers and network providers
2. value chain scenarios and examples of benefits
3. the TV-Anytime environment in terms of persistent storage, a return path, business models and a functionality roadmap
4. consumer-centric usage scenarios

The combination of business models with the functionality roadmap is of particular interest. They are also expected to be used in the specifications document (S-1). The four models are described below, followed by the functionality roadmap

1a Basic 'push' model
(Free TV + Pa TV via uni-directional delivery)
1b Basic 'push' model with conditional access
(Free TV + Pay TV via uni-directional delivery)
2 Consumer-response model
(Free TV + Pay TV with return path)
3 Full-interactive model
(Free + Pay TV with broadband Internet connectivity).

Note that within the myTV project the main focus is on model 2, the consumer-response model.

Feature level	PDR Functionality	Model			
		1a	1b	2	3
1	Impulse record				
2	Record and playback an A/V stream				
3	Simultaneous record and playback				
4	Live pause				
5	Content search and access				
6	Playback of content in indexed/highlight mode				
7	Book-marking & personalization				
8	Portability of consumer profile (personalization/agents)				
9	Storage management – basic & advanced (e.g., quality)				
10	Profile management (personalization/agents)				
11	Multi-user profile support (personalization/agents)				
12	Targeted services using local profiles (personalization/agents)				
13	Updating of content				
14	Multi-modal operation				
15	Compliance to all content delivery mechanisms				
16	Synchronization of stored & 'live' content				
17	Support for a variety of content types regardless of source, delivery channel or medium				
18	Transferring/archiving content to/from other local devices via analog I/O				
19	Using other devices to control PDR remotely				
20	Access to premium content				
21	Transferring/archiving content to/from other local devices via digital I/O				
22	Limited E-commerce				
23	Profile exploitation				
24	Ability to pull content				
25	Synchronization of cross media with pull elements				
26	Ability to exploit usage data (e.g.: rights)				
27	Third-party management of PDR				
28	Updating of content with verification				
29	Content usage verification				

30	Networking using the PDR (e.g.: E-mail, remote control by mobile, etc.)				
31	Full Pay-TV/E-commerce capable				

S-1 "Benchmark Applications" (WD158 – final document will become TV047)

The first in the TV-Anytime specifications series, S-1 is currently a working document which is expected to be relatively brief and which will provide more guidelines for the implementation of various elements of the business models and functionality roadmap.

The document is still in rough draft form, but is expected to contain:

- An introduction to key concepts covered by the specifications
- Description of three pre-defined models and phase and scope
- Specification benchmark features

The current draft includes the business models and functionality roadmap from R-1 as well as benchmark features within the areas of:

Record and playback	(feature level 1-4 from the roadmap)
Content search and access	(feature level 5)
Playback of content in indexed/highlighted mode	(feature level 6)
Personalisation and preferences	(feature level 7 – 11)
Updating of content	(feature level 9)
Multi-modal operation	(feature level 13)
Non-AV content support	(feature level 15 – 17)
Transferring and archiving	(feature level 18)
Networking using PDRs	(feature levels 22-29)

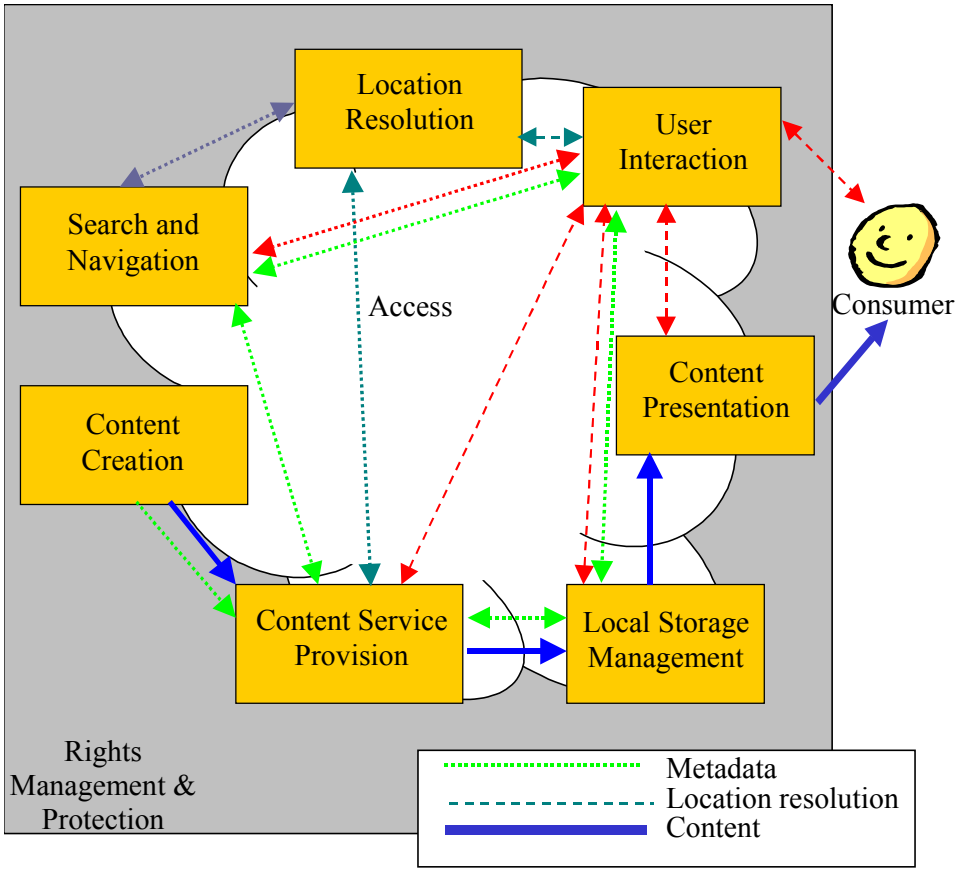
After finalizing the benchmark applications, the business models group will continue to work on the next phases of TV-Anytime.

2.2 System design

The goal of the system design group is to develop an overall system reference architecture for the TV-Anytime system. The architecture should provide a context and common language to enable the development of the TV-Anytime specification. This context is needed to tie together all the tools that are being developed in the TV-Anytime specification. The group is chaired by Frits Klok (KPN), assisted by Ronald Tol (Philips).

The architecture developed in the group tries to accommodate as many different business models as possible. It shows the end-to-end system relationships in a TV-Anytime service. These relationships are described both from a functional and a dynamic viewpoint. A number of interfaces are identified that may need specification in a TV-Anytime context. These interfaces may contain one or more different data flows that carry the actual information.

A simple TV-Anytime 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. Examination of the mechanisms behind this simple model leads to decomposition in a functional reference model that is depicted in the diagram below.



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 access function is depicted as a cloud, and symbolizes a connectivity function between the functional units. This can be anything from a single wire inside a PDR, to a full-fledged broadband network. A rights management and protection system is essential to the system, and is depicted as a grey box that encompasses all aspects of the model. TV-Anytime is working on a way of incorporating rights management protection into the overall system model.

The basic flow of content in the model is indicated with a blue bold line, metadata flows are indicated with a green dotted line, location resolution data is indicated with a blue-grey dotted line, and a red dashed line indicates where the user is directly involved.

The requirements for the system are documented in TV036r2 [2]. This document evolved in working document WD124r3 [7]. A first version, TV048, of the system design specification will be released with the final version of the content referencing specification.

2.3 Content referencing

Content referencing is the final stage of a TV-Anytime system that allows a TV-Anytime equipped device to find the content the user has asked for. The user selects a piece of content, which results in the TV-Anytime equipped device having an identifier for a piece of content that is independent of time and location. Content referencing converts this identifier into one or more locations where the content can be acquired from.

The myTV project has maintained an active participation in the content referencing working group of the TV-Anytime forum. As part of this participation, two documents (TV-Anytime document numbers AN076 and AN098) were submitted as joint myTV documents from all the myTV partners.

Document AN076 describes the basic idea of an identifier (which we called a UPI) which has two parts. The first part is called the authority part and refers to an organisation who creates UPIs. The second part of the UPI is free for the authority to use as they see fit. It is the responsibility of the authority to provide a method for the second part of the UPI to be converted into one or more locations.

The document also describes a grouping identifier called a UGI, which works just like a UPI, but translates into one or more UPIs or UGIs.

The v1.0 specification [9] (agreed at the Geneva meeting in July 2000) provides the basic tools for content referencing in a broadcast environment. The solution chosen is very close to the myTV proposal AN076 except for the fact that the UPI and UGI are combined into one identifier called a CRID (content reference identifier). The specification contains

- The definition of the CRID:
- CRID://<authority>;<data>

  Where;
  <authority> Uses the Internet Domain Naming System to assure uniqueness.
  <data> is a free format string that is URI compliant, and is meaningful to the authority given by the <authority> field.

- A mechanism – the 'Resolver Authority Record' for discovery of the resolution data source associated with respective CRID. (e.g. On receipt of a BBC authority CRID the box needs to find the source of mappings between bbc.co.uk CRID and locators). NB This was effectively the proposal from the Philips myTV members as discussed by the myTV team, document AN098.

- The Resolver Authority Record allows for a resolution authority to act as a secondary resolution authority for third party CRID (e.g. the bbc.co.uk authority may volunteer locations for warner.bros.com CRIDs, if they exist). Again, as rehearsed by the myTV project.

This v1.0 specification has been influenced heavily by the myTV proposals (BBC R&D, Nokia, Philips, NOB, University of Ljubljana, NDS, RAI) along with those from Sony, ARIB and TAO.

As the specification is out for public comment, there now follows a period for submissions of comments and suggestions about the specification. It is hoped that such submissions will help the further development of this specification into a full industrial strength specification. The current document was published on August 11th and the deadline for comments is 15th September 2000. The revised specification incorporating public comments is due on 13th October 2000.

The myTV project will now continue to participate in the content referencing group in the area of specifications for content referencing using bi-directional networks.

2.4 Metadata

The metadata group of the TV-Anytime Forum was created to produce normative requirements (document R-3 [3]) and specifications (document S-3 [8]) on TV-Anytime metadata. The group is chaired by Jean-Pierre Evain (EBU), assisted by Curtis Eubanks (Sony).

The metadata group is working on a TV-Anytime metadata model (which still needs work) and a list of TV-Anytime metadata schemas, mostly based on MPEG-7. The group has agreed that the Forum should not specify an encoding format for broadcast metadata. This is considered the job of the broadcast standards organizations like DVB, ATSC and ARIB. The group is studying the need for a encoding format for broadcast related metadata over the Internet. If TV-Anytime want to make a global standard for metadata, the forum need to define an encoding format (such as XML

files) and specify the overall structure of such files. This work in progress will also influence the structure of the myTV box. Members of the myTV consortium are very active in this process also.

An other important part of the workprogram is content segmentation and EPG metadata. This segmentation proposal has been worked out in the myTV project and submitted to the TV-Anytime forum. During this process it became clear that the MPEG-2 stream as it is specified in DVB (and ATSC and ARIB) is not capable of handling synchronized metadata in the way reqruied by TV-Anytime usage scenarios. Proposals to the MPEG community are in preparation to propose solutions to this problem. This is a direct result of the work in the myTV project.

Requirements

The requirements produced by the metadata group can be found in [3]. Key topic-areas are:

1. Content-description (for example to display programmes in an EPG).
2. Classification (related to 1, but also needed for personalisation for example).
3. Segmentation (how to identify, create and use parts of programmes).
4. Storage management (information to support updates and management of available disk-space).

Specifications

The metadata specifications are currently under development. A provisional version of the specifications will be published in September 2000. The final specifications are planned for the end of November 2000 and will be referred to as document TV049.

Liaisons

Several other organisations are also working in the area of metadata. Examples are W3C and Dublin Core. Where relevant, the metadata group forms liaisons to such organisations to enable the sharing of expertise and to develop a stronger base for the technology. The metadata group has liaisons with:

1. MPEG-7

   Two TVAF-members, Ibrahim Sezan (Sharp) and Gandhi Vaithilingam (Philips Research Briarcliff), function as official liaisons with MPEG-7. Also MPEG-7 has formed a TVAF Ad Hoc group to study TV-Anytime as an application of MPEG-7 and to synchronize the activities in both organisations where relevant . Several members of TVAF are chairs of the ad-hoc group.

An other important activity in MPEG-7 is the Metadata integration group activity. This group is working on the possible integration between MPEG-7 and SMPTE / EBU metadata. This activity is important for a seamless integration from production Metadata to consumer applications as defined in TV-Anytime and myTV.

2. SMPTE (Society of Motion Pictures and Television Engineers)

   SMPTE's work on issues like the metadata dictionary, Sets, Packs and Type's dictionary, a very detailed list of technical, production and user metadata, unique material identifiers and the Key Length Value coding protocol are particularly relevant to TVAF. This is underlined by having Peter Mulder as the official liaison to SMPTE.

Snapshot of progress

After having agreed upon the requirements, focus in the metadata group now is on the development of appropriate technology. Important developments are:

1. The use of XML based technology for defining (XML Schema) and expressing (XML) metadata descriptions.
2. The commitment to try to closely synchronize descriptions with MPEG-7 descriptors (XML) and description schemes to develop the largest possible common core for interoperability:
    a. By considering the adoption of MPEG-7 technology on a case by case basis.
    b. By providing feedback to MPEG-7 where applicable.
3. The definition of a DS (Descriptions Schemes) for EPGs:
    a. Classification of content based on Escort 2.4., an EBU practical guideline for the use of classification Metadata.
    b. This may take into account parental guidance.
4. The evaluation of other DSs, such as (among others):
    a. Recording and playback.
    b. User history.
    c. Preferences.
    d. Navigation and selection.
    e. Media review.
5. Work on segmentation:
    a. How to link descriptions to the segments themselves.
    b. The definition of a segment DS.
6. The evaluation of the role of SMPTE metadata within TVAF:
    a. Translation from SMPTE to XML may be necessary (TVAF will not focus on this).
    b. Some lower level SMPTE-elements may be referred to from descriptions (also relevant for segmentation).
7. The specification of security requirements for the protection of metadata related to both:
    a. Access.
    b. Integrity.

2.5 Rights management & protection

NDS and Philips, as active members of the myTV project, have input some of the fundamental concepts incorporated into the TV-Anytime Forum's view on Rights Management and Protection. In particular, the Content Management and Protection principles that NDS has been promoting in a number of forums were submitted to the TV-Anytime Forum and now form the founding philosophy of how TV-Anytime intends to ensure content rights are protected.

Content Management and Protection focuses on the approach of protecting and managing the rights related to individual content items – a paradigm shift from the traditional Conditional Access view of permitting access to the delivery channel for subscription purposes. This approach is fundamental to the technologies used in the myTV project where content items are stored and replayed and not simply delivered across a secure channel any longer.

CMP is about providing protection to items from source to final usage point, or a sensible sub-set of this route (usually at the consumer end). Specific principles contributed by myTV member influence, include:

- The use of Removable Security Modules (e.g. SmartCards) for providing local authorisation and authentication functionality so that breaches can be addressed without imposing on the CE manufacturers
- The use of Secure Authenticated Channels to carry control and management flows
- Authenticated Key Exchange for distribution of keys in the network hierarchy
- The retention of control over use of the content so that there is no irreversible hand-off to uncontrollable systems

First draft publication of an RMP specification by the TV-Anytime Forum is expected 15 December 2000 with final publication shortly thereafter on the 16th February 2000. Participants within myTV, under the guidance of NDS, are currently formulating an RMP contribution document in the form of a white paper, which it is intended, will promote the myTV position to the forum and help steer the specification.

3 DVB MHP

DVB has been working on the Multimedia Home Platform (MHP) specification since 1997. As usual in DVB, the work is divided between a group within the DVB Commercial Module specifying the user and market requirements and a group within the DVB Technical Module responsible for producing the technical specification. The group in the Commercial Module is called Multimedia Home Platform (MHP) and the group within the Technical Module is called Technical Aspects of the MHP (TAM).

3.1 DVB Commercial Module

The DVB Commercial Module has agreed "Commercial Requirements For Guidelines on Improved Recordability of DVB/MPEG signals" (CM 184).

This document recognizes that there is a market demand for the recording of DVB streams and aims to facilitate the recording of DVB streams on consumer storage devices (disc, tape, solid-state).

CM 184 requests the DVB Technical Module to draft recording guidelines that assist in building such recording devices. The guidelines are expected to describe stream characteristics that facilitate recording and show how these can be used to allocate resources and drive functionality of the recording device. In particular it should:

- Recommend stream characteristics that can improve functional features of recording devices (trick-play, editing etc).
- Provide descriptions of how recording devices can obtain parameters from DVB streams that are needed to allocate and manage critical storage resources.
- Describe methods to record all DVB supported formats (audio, video, SI, CA, subtitling, data broadcast, MHP etc).
- Preferably cause no additional operational burden on broadcasters.
- Support CA and CP systems.
- Be cost effective and future proof.

3.2 DVB Technical Module

In January 2000, the TAM group finished the first 1.0 version of the technical specification which was subsequently approved by the Technical Module and the DVB Steering Board. It was sent to ETSI for publication and has been published by ETSI as TS 101 812 v1.1.1 in July 2000.

The MHP 1.0 specification includes the technical specifications for the first versions of the two basic profiles: Enhanced Broadcasting and Interactive Broadcasting. These profiles provide a Java (DVB-J) based API platform for writing interactive applications as well as other topics necessary for building an end-to-end system like transport protocols, signalling, a security model and authentication infrastructure, etc. The goal has been to produce a truly open and interoperable platform allowing independent development of terminals, networks, services and content. The MHP allows the terminals to be completely independent of the service providers and network operators, thus allowing horizontal markets for the terminals while still leaving reasonably much freedom for implementation and differentiation of the terminal products.

After finishing the 1.0 specification, the TAM group has started working on a 1.1 revision that will include some enhancements to the profiles defined in 1.0, as well as a specification for the third profile called Internet Access. It is expected that the 1.1 specification will be finished in January 2001. The enhancements to the Enhanced and Interactive Broadcasting profiles include an API for communicating with smart cards, the possibility to store locally some frequently used applications,

the possibility to use MHP applications delivered over IP through the interaction channel, an optional profile of HTML, etc.

The Internet Access profile means including a regular Web browser in the MHP terminal and necessary integration of that in to the MHP platform. This should not be confused with the possibility of using the Internet and IP as a communication channel for accessing MHP applications and services, which is already supported in the Interactive Broadcasting profile.

The aspects of using local storage for storage of audio-visual and other content, which is the main topic in myTV, has not yet been included in the MHP. The MHP group in the Commercial module has been discussing some user and market requirements for this, but the technical work within the TAM group for inclusion of these has not started yet. It is expected that this will be a major topic after the 1.1 version of the MHP. By that time hopefully the TV- Anytime forum has been able to produce relevant specifications, so DVB could adopt those and define only the necessary mapping to the DVB system and the integration to the MHP platform. The work done in the myTV project will be of valuable input to DVB when the work on this topic begins.

The TV-Anytime Forum specifications will be independent of the delivery systems and networks used for delivering the services and content. DVB will need to specify the appropriate mappings how the TV-Anytime specifications are applied in DVB systems. These will include at least the following issues:

- Mapping of the content referencing system to the DVB location identifiers
- Specifying the means of transporting the content referencing information in DVB streams
- Specifying the means of transporting the TV-Anytime metadata in DVB streams
- Specifying the integration of the TV-Anytime / local storage functionality into the MHP platform, e.g. necessary APIs for MHP applications to control the storage and playback of stored content, etc.

4 Glossary of terms

Name / Acronym	Description
Agent	A system or software that performs a function as a proxy for a user, such as searching based on a personal profile
API	Application Programming Interface
Application	Software that provides a service or function
Bi-directional	A system that allows a two way flow of content and/or information
CMP	Content Management and Protection
Consumer Profile	Data that represents the interests & preferences of the consumer
Content	The media that is stored on the PDR such as audio-visual, text, images etc
Content creator	The producers of the content
Content owner	An entity that owns the intellectual property rights of the content
Content provider	An entity that acts as the agent for and is the prime exploiter of the content
CRID	Content Reference Identifier
Delivery chain	The system that provides the distribution of content & information from providers to consumers
DSL	Digital Subscriber Line
DVD	Digital Versatile Disk
eCommerce	Money based transaction over electronic delivery chains
EPG	Electronic Programme Guide
Home network	A distribution system within the consumer's local, home environment
Home server	A networked storage device within the consumer's local, home environment
IPR	Intellectual Property rights
MHP	Multimedia Home Platform
MP3	MPEG-1 Layer 3 Audio
Network provider	The entity responsible for distribution infrastructure
PDC	Programme Delivery Control
PDR	Personal Digital Recorder
PVR	Personal Video Recorder
PIN	Personal Identification Number
PPC	Pay Per Choice
PPV	Pay Per View
PSI	Programme Specific Information
Return path	Part of the bi-directional distribution system from the consumer to service provider
Service provider	An aggregator and supplier of content which may include gateway & management roles
SI	Service Information
Smart Card	An IC card that contains data such as personal profile, access keys etc.
SMPTE	Society of Motion Picture and Television Engineers

TVAF	*TV-Anytime* Forum
TV-Anytime devices	Components which comply with *TV-Anytime* specifications and requirements
UGI	Uniform Group Identifier
Uni-directional	A system that allows one way flow of content & information
UMID	Unique Material Identifier
UPI	Uniform Programme Identifier
UPID	Unique Programme Identifier
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USID	Unique Segment Identifier
v-mail	Video mail
WIMP	Windows, Icons, Menus, Pulldown menus
XML	Extensible Mark-up Language

5 TVA Workplan

DRAFT TV-Anytime Work Plan
Phase 1 – Simple Model

Status: 27 July 2000

	Mtg Sched	Web Site Development	NAB	IBC		Business Models & System Description (Informative)	System Specification (Normative)	Content Referencing (Normative)	Metadata (Normative)	Rights Management (Normative)
Sept 1999	27 to 29 Sept, Newport Beach, CA	- Establish web site spec - Request web site host and web master			Sept 1999					
Oct		- Identify web site host - Identify web master	- Reserve demo room in Convention Center for NAB - Contact NAB regarding topics for paper sessions		Oct	Technical Solutions Doc used as starting point for Sys Desc		**Begin preparation of Content Ref CFC**	**Begin preparation of CFC for Metadata**	- Begin preparation of CFC for Rights Mgmt - Broaden consultation with content industries
Nov	29 Nov to 1 Dec, Geneva			Identify papers related to TV-Anytime for IBC	Nov	**Relsease of Sys Desc** for release with CFCs	Begin to develop Sys Spec			

	Mtg Sched	Web Site Development	NAB	IBC		Business Models & System Description (Informative)	System Specification (Normative)	Content Referencing (Normative)	Metadata (Normative)	Rights Management (Normative)
Dec		Web site established (17 Dec 1999)	Identify demos to be shown at NAB, and papers to be given that relate to TV-Anytime (21 Jan 00)		Dec			**- Release of CFC (17 Dec 1999)** **- Begin to develop spec frameworks**		
Jan 2000			- NAB prep mtg - Develop brochure for NAB		Jan 2000			Continue to develop spec frameworks		
Feb	1 to 3 Feb, Sunnyvale, CA				Feb			- Continue to develop spec frameworks - Assess input on requirements		
Mar	24 to 27 Mar. ahg Rights Managemen, Osaka 28 to 30 Mar, Osaka, Japan		- NAB prep mtg - Publicize TV-Anytime demos for NAB	- Reserve demo room at IBC - Identify demos to be shown at IBC	Mar			**Content Ref req's due (10 Mar 00)**	**Metadata req's due (10 Mar 00)**	**Rights Mgmt req's due (10 Mar 00)**
Apr	***NAB*** – *(8 to 13 April, exhibition begins on 10th)*		NAB **- Demo room** - Pointers to demo room in member companies' booths		Apr	**Complete Business Model (R-1) and System Description (R-2) docs (7 Apr 00)**		**Complete Content Ref req's doc R-4 (7 Apr 00)**	**Complete Metadata req's doc R-3 (7 Apr 00)**	**First Draft Rights Mgmt req's doc R-5 (7 Apr 00)** Develop spec frameworks

	Mtg Sched	Web Site Development	NAB	IBC		Business Models & System Description (Informative)	System Specification (Normative)	Content Referencing (Normative)	Metadata (Normative)	Rights Management (Normative)
May				Determine if NAB brochure should be updated for IBC	May			**Content Ref technology inputs due (26 May 00)**	**Metadata technology inputs due (26 May 00)**	
Jun	30 May to 1 Jun, New York 4-6 Jun, ahg Metadata, Briarcliff				Jun	Continue to develop business models		Develop Content Ref spec and receive comments	Develop Metadata spec and receive comments	**Second Draft Rights Mgmt req's doc R-5 (16 Jun 00)**
Jul	25 to 27 Jul, Geneva				Jul	Approval of **Complete Business Model R-1 (27 July 00)**	Develop **Version 1 Sys Spec S-2 (27 July 00)**	Approval of **provisional spec S-4 for Content Ref (27 July 00)**		Develop Rights Mgmt spec
Aug					Aug	Release **Complete Business Model R-1 (11 Aug 00)**		Release **provisional spec S-4 for Content Ref (11 Aug 00)**	Develop spec	
Sep	***IBC*** – *(7 to 12 Sept)* 26 to 28 Sept, Los Angeles, CA			IBC **- Mini-conference and demo**	Sep	Approval of **Version 1 of Bench marks Spec S-1 (28 Sep 00)**	Approval of **Version 1 of Sys Spec S-2 (28 Sep 00)**	Receive **comments on Content Ref spec S-4 (18 Sep 00)** Approval of **Version 1.0 Content Ref Spec S-4 (28 Sep 00)**	Approval of **provisional spec S-3 for Metadata (28 Sep 00)**	Approval of **R-5 (28 Sep 00)**

	Mtg Sched	Web Site Development	NAB	IBC		Business Models & System Description (Informative)	System Specification (Normative)	Content Referencing (Normative)	Metadata (Normative)	Rights Management (Normative)
Oct					Oct		Release **Version 1 of Sys Spec S-2 (13 Oct 00)**	**Final publication of Version 1.0 Content Ref Spec S-4 (13 Oct 00)**	Release **provisional spec S-3 for Metadata (13 Oct 00)**	Release **Complete Rights Mgmt req's doc R-5 (13 Oct 00)**
Nov	29 Nov to 1 Dec, Seoul				Nov		Approval of **Version 2 of Sys Spec S-2 (30 Nov 00)**	Begin work on revision	Receive **comments on Metadata spec S-3 (20 Nov 00)** Approval of **Version 1.0 Metadata Spec S-3 (30 Nov 00)**	**Rights Mgmt technology inputs due (20 Nov 00)**
Dec					Dec		Release **Version 2 of Sys Spec S-2 (15 Dec 00)**		**Final publication of Version 1.0 Metadata Spec S-3 (15 Dec 00)**	Develop Rights Mgmt spec and receive comments
Jan 2001	30 Jan to 1 Feb, Washington, DC				Jan 2001				Begin work on revision	Approval of **provisional spec S-5 for Rights Mgmt (1 Feb 01)**
Feb					Feb					Release **provisional spec S-5 for Rights Mgmt (16 Feb 00)**

	Mtg Sched	Web Site Development	NAB	IBC		Business Models & System Description (Informative)	System Specification (Normative)	Content Referencing (Normative)	Metadata (Normative)	Rights Management (Normative)
Mar	27 to 29 Mar, Geneva				Mar		Approval of **final Version of Sys Spec S-2 (29 Mar 01)**			Receive **comments on Rights Mgmt spec S-5 (21 Mar 01)** Approval of **Version 1.0 Rights Mgmt Spec S-5 (29 Mar 01)**
Apr	**NAB** (21 to 26 Apr)				Apr		Release **final Version of Sys Spec S-2 (13 Apr 01)**			**Final publication of Version 1.0 Rights Mgmt Spec S-5 (13 Apr 01)**
May	5-7 June Seattle				May		Begin work on revision			Begin work on revision
Jun					Jun					
Jul	31 July to 2 Aug Tokyo (?)				Jul					
Aug					Aug					
Sep	25 to 27 Sep New York (?) **IBC**				Sep					
Oct					Oct					
Nov					Nov					
Dec	4 to 6 Dec Geneva				Dec					

6 References

[1] R-1 Business Model Requirements Document (Informative), TV035r6, TVAF.
[2] R-2 System Design Requirements Document (Informative) , TV036r2, TVAF.
[3] R-3 Metadata Requirements Document (Normative), TV037r3, TVAF.
[4] R-4 Content Referencing Requirements Document (Normative), TV038r2, TVAF.
[5] R-5 Rights Management Requirements Document (work in progress) (Normative), TV039r1, TVAF.
[6] S-1 “Benchmark Applications”, TV047, TVAF
[7] S-2 System Design Specification TV048, TVAF
[8] S-3 Metadata Specification Document TV049, TVAF
[9] S-4 Content Referencing Specification Document TV050, TVAF

Appendix A : AN032

Included below is the text from a proposal/contribution to the TV-Anytime Forum, jointly made and submitted by myTV project partners in January 2000. It suggests further requirements for content referencing.

The requirements for content referencing set out in the CFC document released by the TV-Anytime Forum on December 17, 1999 form a good starting point for building an attractive TV-Anytime system. We would like to suggest that as a next step we again look at attractive TV-Anytime features and application scenarios from the user's perspective, and see if there are any that the current set of requirements cannot support. To start off this process, we have collected some feature-based scenarios, quite a few of which are already covered by the current requirements.

EPG

The user presses a button and an electronic programme guide for today's television programmes appears. As well as the normal transmission time and title that a conventional EPG contains, it also has synopsis information, genre, cast list and links to other related content (either also on TV or on the internet).

When the user selects a programme to record, the storage device will store an identifier for the programme, which will be resolved to a time and date of transmission later on. The storage device will be able to record this programme even if it moves from its scheduled transmission slot.

SEGMENTATION

Whilst out of the room, the viewer of a football match misses that all-important goal. By the time he returns to the room the replays sent by the broadcaster have finished. With his TV-Anytime equipped time-shift device, he can press a button and return to a point just before the goal.

FILM CRITIC'S WEB SITE

Whilst surfing Roger Ebert's film review site, our film fan user finds something she would like to record as the web site gives it a good review and it is supposed to be broadcast in the next few weeks. She clicks on a special link, and the storage device is automatically programmed to record this film. The box does not know when the programme will be transmitted yet, nor on which channel. It will find this information out closer to the transmission date when the TV schedules have been fixed.

TRAILER FOR A SERIES

A trailer is shown for a series that will be shown this season. The user selects this trailer, and the TV-Anytime equipped storage device stores an identifier for this series. When the series is broadcast, the stored identifier allows the storage device to record each episode in the correct order. If episodes are repeated, the storage device can use this to avoid conflicts with other recording requests it has received.

A FILM IN TWO PARTS

A film has been selected for recording, but when transmitted there is a 40-minute gap in the middle of the film where the broadcaster has placed their regular nightly news bulletin. The TV-Anytime information allows the storage device to record the film without the news bulletin in the middle, or just the bulletin.

A PROGRAMME LIKE THIS

After watching a natural history programme about Kenya's national park, a family viewing this programme decides to find out more about the wildlife of Africa. Using their TV-Anytime equipped storage device, they can tell the box to record some programmes similar to the one they have just

watched. They can also use the embedded URLs in the TV-Anytime information to visit web sites directly connected to the programme they have just seen.

SPLIT CHANNEL TRANSMISSIONS

A tennis tournament is being shown live on channel 1 until a certain time, when it will continue on channel 2. A viewer has decided to record one specific match as they are following one particular player though the tournament. The TV-Anytime equipped storage device knows when the broadcast starts, and can then wait for the desired match to start. During this match the transmission moves to channel 2, which the storage device follows by using the transmitted TV-Anytime information.

Appendix B : AN070

Included below is the text from a proposal/contribution to the TV-Anytime Forum in March 2000. It suggests further requirements for metadata.

The current requirement document on metadata lack a few requirements that possibly make it harder to develop a TV-Anytime system also in the analogue or hybrid domain. Also we think that it is not sufficiently clear in the current document that all types of content including adverts, trailers, etc can have metadata attached. We, therefore, propose to add the following requirements.

- A TV-Anytime metadata scheme must be mappable on currently available metadata schemes for analogue TV (e.g., NextView in Europe).
- A TV-Anytime metadata scheme must provide fixed or selectable defaults to enable the use of non attributed audio/video coming from analogue TV.
- The metadata scheme should make a high volume / low cost implementation(esp in the analogue domain) possible. This might require several profiles of metadata or a minimum core set with extensions or ...
- All types of content including adverts, promos or trailers, parts of programmes, etc. can have descriptive metadata attached.

Appendix C : AN076

Included below is the text from the proposal/contribution to the TV-Anytime Forum, jointly made and submitted by the myTV project partners in March 2000. It deals mainly with content referencing, but also touches on meta data.

Introduction

This document contains a proposal for a content referencing and location resolution framework for TV-Anytime that is independent of the delivery system. To help to clarify this proposal, examples are provided for a mapping to DVB.
This document defines a hierarchical content referencing system which is independent of time and location of the actual content. We introduce group nodes and terminal nodes. Group nodes can be used to group different content together, for example a Star Trek series. A terminal node only refers to one piece of content, for example a Star Trek episode, but may resolve into multiple locations (in time and space) for this piece of content. A group node will resolve into one or more terminal nodes, which in turn will resolve into one or more occurrences of the content.

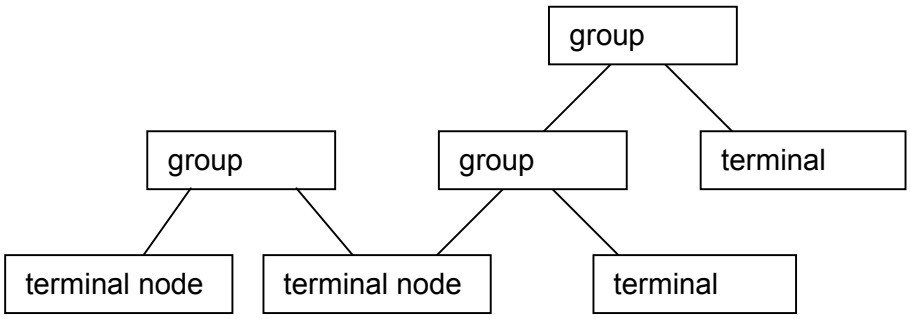


Figure 1 Hierarchical content referencing example

Each of the nodes can be pointed to with a Uniform Resource Identifier (URI). These URIs should identify the piece of content denoted by the node and be independent of the location and time of where and when this piece of content can be obtained. The terminal nodes can be resolved into the location and time bound information that again can be described with an URI (a URL in this case).

The information required for TV-Anytime functionality includes the following:
1. LOCATION RESOLUTION; reveals where and when a particular programme can be obtained.
2. GROUP RESOLUTION; reveals which programmes belong to a particular group such as a serial.
3. CONTENT INFORMATION; gives descriptive information about content such as synopsis, genre, etc.
4. SEGMENTATION INFORMATION; reveals the beginning and end positions of programme segments such as items in a news bulletin.

Content referencing

Crucial to the concept of TV-Anytime is the ability to refer to content, say a television programme, uniquely and independently of its channel and time of broadcasting.

The UPI

We define the Uniform Programme Identifier, or UPI. This is the URI scheme used for the terminal nodes. It takes the following shape:

upi:<authority>:<unique code>

Figure 2: Format of UPI

A UPI is a terminal node and so can only refer to one piece of content. A UPI may resolve to multiple locations however.

The UGI

The URI scheme for the group nodes is called the Uniform Group Identifier or UGI, and is defined in a similar manner:

ugi:<authority>:<unique code>

Figure 3: Format of UGI

For both UPIs and UGIs, the <unique code> part is to be unique within the name space of the 'authority', but not globally unique. This allows for resolution authority to be in charge of its own upis, and no global institution is required to hand out unique codes. Naturally the 'authority' needs to globally uniquely identify that authority. It is also permissible for different authorities to assign different UPIs to the same content.

The Resolution Authority

The 'authority' part of the UPI and UGI will need to be globally unique. One method to assure this would be to create a registration authority and require all resolving authorities to register with this TV-Anytime authority. Creating and running such an organisation may not be desirable, and so this document proposes to use the DNS registration system to ensure uniqueness. The <authority> part of UPI would be a valid DNS name and optional path.

bbc.co.uk
raiuno.rai.com
www.yahoo.com/fredbloggs

Figure 4: Examples of UPI authorities

Ideally, the same UPI/UGI can be used to identify a piece of content independently of where it will be available for retrieval, e.g. whether it will be broadcast on a broadcast service or if it is available for download from an Internet server. The usage of the DNS registration system as the authority codes does not imply necessarily that the content will be delivered over the Internet. These UPIs can be used for identifying pieces of content that are available e.g. only via broadcast.

The URL

Each UPI will need to resolve into one or more locations in time and space for the piece of content. A UPI resolving to multiple URLs allows the PDR to perform recording conflict management.
As these URL formats are typically specific to a delivery mechanism, we do not expect the TV-Anytime Forum to specify the syntax of them. However, the following requirements need to be fulfilled:

- identify the transport mechanism
- identify the location in space with appropriate means for the transport mechanism
- if necessary (e.g. for broadcast systems), identify the (approximate) time when this piece of content will be transmitted

To allow a PDR to be able to identify the transport mechanism, the URL formats proposed in this document all conform to the following form:

<system>://<location>[@<date>T<time>D<duratio n>]

Figure 5: URL format for delivery systems defined in this proposal

As TV-Anytime is unlikely to specify all possible formats of URL, the constraint given in Figure 5 may not be achievable for all delivery systems.
The <system> part defines which transportation system to use, for example dvb, http, analogue tv. The notation used for the optional items of <date>, <time> and <duration> is in line with ISO8601. This time indication is used for showing schedules in user interfaces, and for approximate timing of recordings in a push system. Push systems (e.g. broadcast) typically have their own means of signalling exact start and end times of the event.
For content sent over a DVB system, the URL format is an extension of the format specified in part 9 of the DAVIC specification:

dvb://<original_network_id>.[<transport_stream_id>].<service_id> [.<component_tag>][;<event_id>]@<start_date>T<start_time>D<durati on>

Figure 6: Format of URL for DVB

For content sent over an analogue TV system, the URL formats are:

PAL://<channel>[.<PDC code>]@<start_date>T<start_time>D<duration> SECAM://<channel>[.<PDC code>]@<start_date>T<start_time>D<duration> NTSC://<channel>@<start_date>T<start_time>D<duration>

Figure 7: Format of URL for analogue TV

Resolution process of UGIs and UPIs

The UGIs need to be resolved into UPIs and the UPIs further to URLs to be able to actually locate the piece of content. This requires some resolution mechanism to provide this.
There are basically two basic ways how this can be achieved:

- the mapping information for UGI-UPI and UPI-URL is delivered via broadcast (push model)
- the device sends the UGI/UPI to a server that performs the resolution and returns the result (pull model)

We believe that both of these mechanisms should be supported.
When getting a UGI or a UPI, the device needs to be able to find where this resolution information (either push data or a pull server) is found. This is done with the authority code.
When sending the resolution information over broadcast, the services that carry this information can indicate for which authority (or authorities) the information is sent within that service. Then the device can locate the right service where the resolution information for that given authority is carried in.
When using a server in the Internet to provide a resolution service, the Domain Name Service for the domain indicated in the authority part of the UPI could be used to discover the name of the resolution server. Naturally, a protocol needs to be specified how to communicate with this server to perform the resolution.

Metadata

As part of a TV-Anytime solution it will be necessary to be able to connect content references to their respective metadata. The ability to connect metadata to content is achieved in this proposal by using the UPI or UGI as the key in a metadata database.

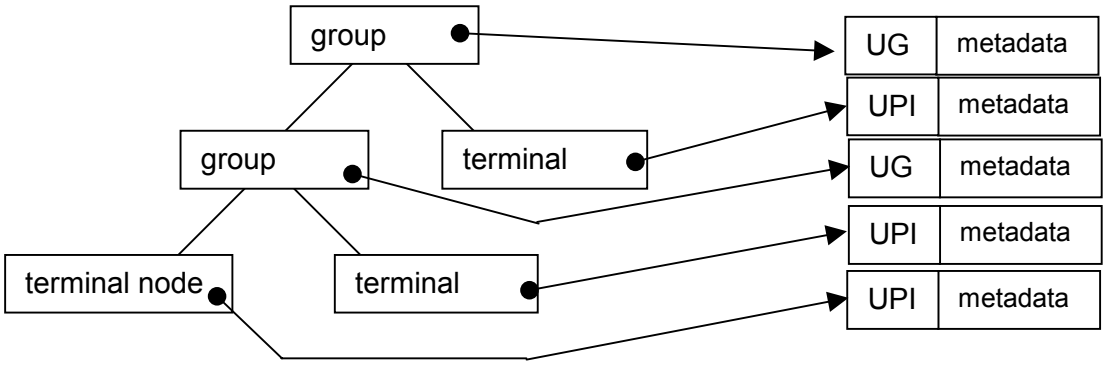


Figure 8: Linkage of Metadata to UPI/UGI

UPI or UGI	Metadata
upi:digital-video.org.uk:smoke	title=”Dying for a Smoke”, genre=”comedy” ….
ugi:bbc.co.uk:AllNewsBBC1	title=”All News on BBC1”, genre=”news”, ….
ugi:www.barrynorman.com:Weekly 10	title=”Top 10 films on TV this week”, …..

Figure 9: Example of metadata linkage

Profiling

Several TV-Anytime business model examples require the ability to create a profile of the user (which may be subject to privacy legislation and/or possible contracts with the user). One simple method of achieving this would be a keep a list of the UPIs that a user has viewed. The hierarchical system proposed here can be used to improve this simple list of UPIs by providing grouping.
A PDR can use the UGIs that it receives to convert some UPIs into UGIs when the UGI contains UPIs that are in the user’s profile. For example, a user who has watched the first two episodes of a series will have the UPIs for these episodes in their profile. The PDR can replace these UPIs with the UGI for the series (as the list of UPIs will match) and then record the rest of the series automatically by just using the UGI.

Transportation of TV-Anytime Data

The TV-Anytime data will need to be transferred over a variety of networks ranging from uni-directional broadcast to bi-directional switched. Some networks might provide near error free transmission whilst other networks might have long latencies or numerous errors (e.g. mobile telecommunication). The proposal outlined in this document is to use one common data format which can be translated to other forms for transportation to the PDR in such a way so that it can be converted by the PDR back to this common format.
Any system that can access a network will already have the capability to decode the protocol used on the network, so any TV-Anytime implementation that fits within these ‘native’ protocols will be cheaper to implement than a system using a completely new protocol. Additionally, some of the information needed for TV-Anytime functionality is already present in many systems in different forms, so it is not a good idea to duplicate this information unnecessarily.

The idea of this proposal is that the TV-Anytime equipped storage device can convert the protocol on the network back to the common TV-Anytime data format, if necessary.

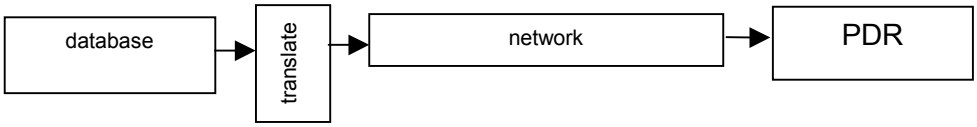


Figure 10: Simple Transport example

A PDR device with access to multiple networks can receive TV-Anytime information from any of its networks and yet combine all the information it gets.

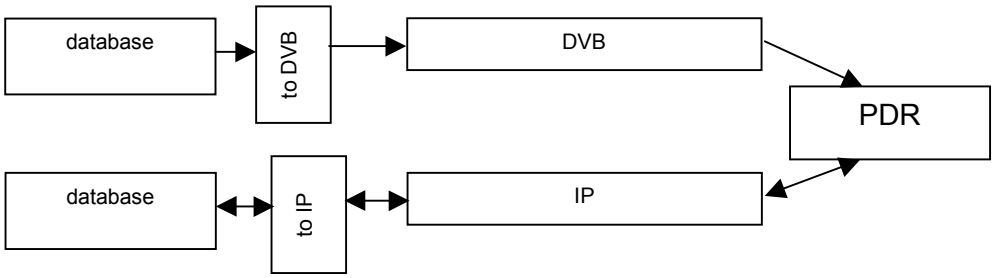


Figure 11: Transport Example with Multiple Networks

Examples of UPIs, UGIs and URLs

upi:nob.co.nl:12345
upi:bbc.co.uk:1317419209
upi:sky.com:2103001930

Figure 12: Examples of UPIs

ugi:bbc.co.uk:B1Eneastenders
ugi:imdb.com:Top10ActionFilms

Figure 13: Examples of UGIs

dvb://bbc1;1a639f@19991129T201500D013000
http://yahoo.com/fredblogs/my_wedding.vob
PAL://NED3.514155@20000320T1830D0030

Figure 14: Examples of URLs

An XML syntax for Location Resolution

This appendix provides a possible format for the common format for TV-Anytime location resolution.
In the following section, items inside XML definitions that indicate where data should be placed are shown in *italics*. For example

```
<PROGLOC ID="UPI">
```

The item inside quotes of ID="*UPI*" would be replaced with a UPI identifier.

Location Resolution format

This format gives the locations where content can be obtained, i.e. it links UPIs to URLs.

```
<LOCATION_RESOLUTION>

<PROGLOC ID="UPI">
list of URL locations, each one inside a location tag
</PROGLOC>

</LOCATION_RESOLUTION>
```

Figure 15: Syntax of location_resolution tag

The example below should be sufficiently self-explaining. It contains only one programme, which is repeated the next morning.

```
<LOCATION_RESOLUTION>

<PROGLOC ID="upi:bbc.co.uk:1317419966">
<location url="dvb://bbc..bbc2;19990916T1501D0028"/>
<location url="dvb://bbc..bbc2;19990917T0901D0028"/>
</PROGLOC>

</LOCATION_RESOLUTION>
```

Figure 16: Example of location resolution tag

Group Resolution format

This format reveals the composition of programme groups.

```
<GROUP_RESOLUTION>

<GROUP ID="UGI"
TITLE="title text"
COMPLETE="TRUE or FALSE" ORDERED="TRUE or FALSE"  REPLACE="TRUE or
FALSE">

list of UPIs or UGIs, each one inside location tags
</GROUP>

</GROUP_RESOLUTION>
```

Figure 17: Syntax of group_resolution tag

The TITLE attribute of the group tag provides the title for this group. The COMPLETE attribute defines whether this group is complete, or should be resolved again later on as new items will be added. The ORDERED attribute defines whether a group has an order (e.g. episodes of a drama series) or is unordered (e.g. a news programme). The REPLACE attribute controls whether each item in the list should overwrite previous items in the group when recorded. An example of a group with REPLACE=”TRUE” might be a news programme where the last news programme replaces the previous one.

```
<GROUP_RESOLUTION>

<GROUP ID="upi:bbc.co.uk:readycook"
TITLE="BBC2 series: Ready, Steady, Cook"
COMPLETE="FALSE" ORDERED="FALSE"  REPLACE="FALSE">
<location upi="upi:bbc.co.uk:1317419209"/>
<location upi="upi:bbc.co.uk:1317419966"/>
<location upi="upi:bbc.co.uk:1317420220"/>
</GROUP>

</GROUP_RESOLUTION>
```

Figure 18: Example of group_resolution tag

Figure 18 shows an example of a group of programmes. Three are listed, which is part of a series, but there are more to follow as the complete field is set to false. The order in this series is irrelevant as ordered=false and the episodes do not replace one another as news bulletins would (because replace=false).

Appendix D : AN098

Included below is the text from the proposal/contribution to the TV-Anytime Forum, jointly made and submitted by the myTV project partners in May 2000.

Introduction

Before a PDR can resolve a UGI or UPI, it needs to know where to find its location resolution provider. In the myTV proposal to the Osaka meeting (AN076), a DNS name was specified as part of the UPI and UGI which was called the "resolution authority". However, there was no detail on how this DNS name is used by the PDR to find its resolution information. This document describes a solution to this problem.

Resolving Authority Record

Each resolution authority (i.e. DNS name used in either a UPI or UGI) will require one or more resolving authority records to exist in the PDR for location resolution to take place. Each record consists of:

Authority name	The DNS name used in a UPI or UGI
URL	The URL where resolution information can be found
Class	Primary or secondary
First valid date	The first date when this authority can be used (using UTC as time ref)
Last valid date	The last date when this authority can be used (using UTC as time ref)
Weighting	Used for ordering multiple resolution records.
Descriptive text	Optional text to describe this authority

Figure 1: Resolving Authority Record

The URL could point to a broadcast stream, or to a server on the internet. In the case of multiple records for the same authority, the PDR can choose to just use one of them, or try them all in turn. The weighting field can be used to give a hint to the PDR as to the order to try multiple records by providing the lowest weighting number to the resolution provider that should be tried first.
The class field defines whether this authority record defines a resolution authority which can resolve all UPIs or UGIs for this authority name (class = primary) or only resolves some UPIs or UGIs for this authority name (class = secondary).
The reason for providing start and end dates for resolution is so that resolution providers can move their resolution URLs and be sure all PDRs have switched to the new URL once the last valid date of the old resolution record has passed.
Here is an example:

Authority name	bbc.co.uk
URL	dvb://bbc1.1a63.09
Class	Secondary
First valid date	2000/05/15
Last valid date	2000/06/30
Weighting	1
Descriptive text	The BBC's TV-Anytime service on DTT

Figure 2: Example Resolving Authority Record

The URL for the above example is similar to the URL given in the myTV proposal AN076, and contains original network ID, transport stream ID and service ID. It is merely an example of a URL

format for a DVB system, and is not meant to define a format that must be used for all broadcast systems.

Authority name	bbc.co.uk
URL	http://www.bbc.co.uk/tvanytime/resolution
Class	Primary
First valid date	2000/02/01
Last valid date	2000/10/31
Weighting	2
Descriptive text	The BBC's TV-Anytime service on the internet

Figure 3: Example Resolving Authority Record

When a PDR receives a UPI or UGI for bbc.co.uk, it will try the DVB stream bbc1.1a63.09 and if this fails and it has an internet link, it will then try http://www.bbc.co.uk/tvanytime/resolution.
It is beyond the scope of this document to describe how an internet URL is used for location resolution. The URL could point to a database to download or point to a CGI script where the UPI / UGI to resolve forms part of the http request (e.g. http://www.bbc.co.uk/tvanytime/resolution?UGI=bbc.co.uk:EastEnders).

Service Advertising

A broadcaster needs a way to push authority records onto the PDR so that the PDR knows where to look in the broadcast stream for location resolution information. This document proposes adding resolution service advertising information to the broadcast stream. It would need to be available on all TV channels a broadcaster provides, as there is no way for the PDR to know where to look for resolution service advertising information.
As an example, in DVB it could be carried in each transport stream in either the NIT or BAT, as this would allow a PDR to find the service advertisement regardless of the TV channel it was tuned to. The resolution record is quite small and so could be placed inside a DVB descriptor in the NIT or BAT quite easily.

Non-broadcast Resolution

A party that wants to provide location resolution, but does not have access to the broadcast chain requires another method to allow the PDR to get their resolution records.
RFC 2052 provides an expansion of the DNS system that is currently used to allow internet connected machines to find mail servers. Rather than just being able to search for mail exchange (MX) records, it is now possible to search for service (SRV) records. SRV records are supported by the UNIX DNS server bind v8.1.1 and above and by Windows DNS v5.0 and above. We would need to decide on a name for the TV-Anytime service record, for example resolution.tvanytime.
If an organisation without access to the broadcast chain wishes to provide location resolution, the DNS name they use for their authority name must point to a DNS server that can return an SRV record that points to a server that will provide location resolution information.

E.g.
upi:barry-norman.com:thegreatescape

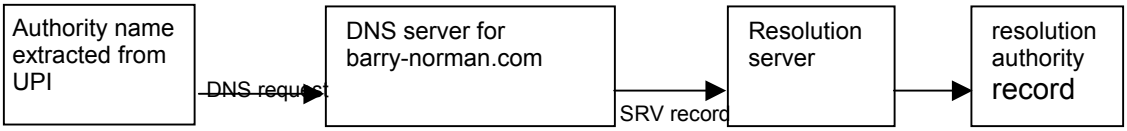


Figure 4: Resolving using DNS

The PDR does not have a resolution authority record for barry-norman.com, but it does have an IP link. The PDR issues a request to the DNS server for “barry-norman.com” for an SRV record of “resolution.tvanytime.barry-norman.com”.
The machine address returned by the SRV request can then be contacted for a resolution authority record. This resolution authority record can then be added to the PDR’s resolution authority database, and resolution of the above UPI can be carried out just as in the broadcast case given earlier.

Appendix E : AN110

Included below is the text a proposal/contribution to the TV-Anytime Forum, primarily made by NOB, one of the myTV project partners, in May 2000. It deals with some first thoughts on segmentation.

First thoughts on extension of the myTV UPI-concept to facilitate segmentation and absolute content referencing.

Summary

This document describes a proposal for the extension of the myTV UPI-concept to allow it to be used for segmentation and absolute referencing at any desired level of detail (up to frame-level). It starts with a general overview of segmentation and a list of example services illustrating the need for an absolute content-reference identifier. Following the USID is introduced, as a way to extend the myTV UPI-concept to segment-level. The proposed USID can easily be constructed starting with the UMID, the standard material identifier proposed by SMPTE. Also some ideas on the use of USID-related metadata and viewing-paths ('playlists') are discussed. This document is meant as a first proposal. A number of open issues are identified in chapter 5 to facilitate discussion.

E-0 References, abbreviations and general remarks

Some referenced or otherwise related documents:

[1] AN032, Suggestions for further requirements on content referencing, January 2000.

[2] AN076, myTV Proposal on TV-Anytime Data for Content Referencing, March 2000.

[3] TV037r2, Requirements Series R-3 on Metadata Requirements, 7 April 2000

[4] TV038r2, Requirements Series R-4 on Content Referencing Requirements, 7 April 2000

[5] SMPTE 330M, Unique Material Identifier (UMID) - Proposed SMPTE Standard for Television.

[6] SMPTE 312M, Splice Points for MPEG-2 Transport Streams – SMPTE standard.

[7] DAVIC 1.5, revision 6.0, TV-Anytime and TV Anywhere.

[8] SMPTE 298M, Universal Labels for the Unique Identification of Digital Data.

Abbreviation	Meaning
ISP	Internet Service Provider
SEGMET	SEGment METadata
UGI	Uniform Group Identifier
UMID	Unique Material Identifier
UPI	Uniform Programme Identifier
URL	Uniform Resource Locator
USID	Unique Segment Identifier
WORM-TC	Write Once Read Many – Time Code
32	Indicates number of bytes used (32 bytes in this example)
3b	Indicates number of bits used (3 bits in this example)

Broadcaster: While talking about ‘broadcasters’ we do not limit its definition to traditional broadcasters; the content might also be obtained via an ISP (as a file containing an AV-programme for example or streamed via the Internet).

Sharing: Exchanging interesting information, links, bookmarks, etc. with other myTV-users (for example via the Internet, within the same box, etc.)

E-1 Introduction

The new functionality and services introduced by the TV-Anytime system are built on the assumption to be able to reference content unambiguously and at detailed level. Also there is a need to be able to divide material into appropriate segments (the definition of 'appropriate' depends on the service/functionality).

This document discusses issues related to both absolute content referencing and segmentation. While a number of standards for the access to broadcast streams already exist, we feel there is a need for a description on a higher (conceptual) level. Existing techniques may be used to implement this functionality. We will touch upon implementation-issues to make the reader aware of possible practical implications. In general however, the exact implementation of the mechanisms discussed is outside the scope of this document, making the concept universal and network-independent.

Note that the term 'content-referencing' also is often used in discussions on the UPI-UGI concept. This document basically extends that concept to within-programme level ('extends', because the UPI could be used as part of the referencing mechanism within a programme). Also the document introduces a link to SMPTE's UMID, to ensure that future implementations at the broadcaster's side will conform to standardisation practices. An issue that is also important when metadata-filtering techniques based on SMPTE-standards are used in future consumer-devices.

E-2 Segmentation

2.1 Programme-hierarchy

Although the TV-Anytime system may support various kinds of content, ranging from audio-visual content to data-services, in the following we will focus on audio-visual (television) programmes.

Before discussing segmentation, let us introduce a general hierarchy for AV-programmes. Table 1 presents a programme-hierarchy, dividing programmes in seven levels. Level 1 is the highest level, comprising of programme-groups that each consists of programmes that are related in some way (e.g. same broadcaster, same theme, …). The second level is the AV-programme itself, which may be broken down into different items*, representing level 3. The notion of items is appropriate for some broadcasts (like news-programmes), but the name may be less intuitive for others (a romantic movie for example). Anyway the idea is that items form a coherent group of scenes and thus form an intermediate level between programmes and scenes. Scenes themselves consist of multiple shots and typically share the same setting (e.g. background). At shot-level (level 5) the definition is relatively easy, while shots can be defined as an uninterrupted recording of AV-content (or as a piece of content produced without pausing/stopping the camera). The frame-level can be defined as the basic building-block of video-sequences, consisting of a single, full-resolution image taken at a single time-instant**. Within a frame one may be able to distinguish between regions or objects.

We like to remark that the foregoing 'loose' definitions are only meant to define a common reference-frame. Also note that the definitions are of various types. Frames and shots can be regarded as very technical definitions, while items and scenes are much harder to define without interpretation of the content.

Level	Name
1	Programme group
2	Programme
3	Item
4	Scene
5	Shot
6	Frame
7	Region/object

Table 1 Programme-hierarchy for AV-programmes.

Segmentation in general can now be defined as segmenting any of the higher-level (lower numbers) elements into lower level ones. For levels 1 and 2 this is handled by the myTV UPI-UGI scheme, while for the other levels within myTV no mechanism has been defined yet. In the following we will assume the goal is to segment programmes (level 2) into segments at any of the levels 3-6. The argument for having multiple 'goal-levels', is that scenarios can be thought of where other than item-level segmentation might be desirable, for example building a scrapbook of all appearances of the viewer's favourite actor (shot-level).

** We deliberately do not use the word segment here, because we will use it to identify segmented content in a more general way.*
*** A field might be seen as an even more basic building block, but it is at half resolution compared to a frame.*

Although this idea could be extended to level 7, this proposal does not deal with it. A second remark here relates to the use of the term 'segmentation'. While segmentation can be seen as "chopping" a programme in pieces, this document mainly addresses the need to be able to go to (the beginning) of these pieces. Indexing might be a better word for this functionality. In the remainder we use the word segmentation to refer to both, as every index can be thought of as a link to a segment.

2.2 Service-requirements

So for what reasons is segmentation needed? Following is a list of functionality a consumer might wish to be able to use with regard to programme-segmentation. Examples are provided for all services. We distinguish between explicit use of segmentation and implicit use of segmentation. The first refers to a situation in which the consumer is relatively aware of 'segmenting the programme, while in the latter the segmentation is buried 'deeper within the service'. Of course the border between the two must be seen as illustrative, rather than perfect.

2.2.1 Explicit use of segmentation

1. Labelling interesting events within a programme with a bookmark.
   The consumer may want to build a scrapbook of his favourite sportsman or mark all appearances of lovely islands.

2. Storing, sharing and reusing bookmarks.
   Because the consumer is part of a group of sportsfans, he likes to show the sportsman's appearances in his recorded material to his friends later (storing), wants to e-mail them to his friends who have recorded the same material but didn't segment it (sharing) or he wants to review the appearances (reusing), possibly extended with those received from his friends.

3. Self-initiated non-linear viewing of a linear programme.
   Although the broadcaster did not provide the consumer with segmentation-information, the consumer wants to skip certain parts and view them in a different order. He uses the VCR-functionality (rewind, fast-forward, skip N seconds, jog-shuttle). This can be seen as an enhanced version of 1, where the use of the VCR-functions may build a list of indexes 'implicitly' (for example not storing the use of 'skip N seconds', except when it is followed by the 'play'-command).

4. Storing, sharing and reusing a non-linear viewing path.
   Similar to 2 the consumer wants to be able to store, share and reuse his non-linear viewing path.

5. Editing a programme.
   Editing can be seen as a more explicit use of segmentation, compared to 1 and 3. The consumer now is totally aware he is reordering the program and definitely wants to keep (and maybe share) the results as a new 'content-package'. The old content might be discarded completely (note that in 3 this may be undesirable). The material used for editing could come from various sources ('home-made', recorded broadcasts, web-content, etc.). The reasons for having editing-capacities might be artistic, but editing may also be used as a form of parental guiding.

6. Programme-segment index
   The consumer is provided with an index of programme-elements he can watch. He also might be allowed to search the index-information.

7. Segment recording
Instead of recording a complete programme, the consumer may want to record part(s) of it only. This could be beneficial with regard to saving disk-space.

2.2.2 Implicit use of segmentation

8. AA-news service
For the 'Always Actual'-news service segmentation is needed to be able to update items instead of the complete programme.

9. Commercial skipping/targeted adds
Commercial identification can be seen as a form of segmentation also. Suppose a programme is broadcast containing four commercials. The consumer may be allowed to skip two of them or the viewing-order changes depending on his personal preferences. Anyway if the commercials are part of the programme, segmentation-information is needed to be able to identify them. Note that this segmentation of course needs work to be done by the broadcaster. If the broadcaster decides not to enable commercial-skipping, one way of achieving this would be to not identify the commercials. The consumer then will be forced to view the first part (seconds?) of the commercial before he might try to use VCR-controls to skip it. The latter may be disabled by the box also.

10. Play-along games
For play-along games several issues may be important. Segmentation-information may be needed to be able to reorder/skip segments (depending on the correctness of the consumer's answer for example). Also the synchronisation of textual-overlays and other local interactivity needs segmentation for synchronisation with the broadcast programme. Note that the resulting 'viewing-path' might have to be stored for later (re)play.

11. Viewing material in a non-linear way, as provided by the broadcaster.
This service is related to 3, but now the consumer may be less aware of non-linear viewing being used. When viewing a documentary on whales which is provided with links to related chapters and an index of all whale-species, the viewer may perceive this as a special experience in which the concept of a 'linear underlying programme' is not noticed (in 3 this is noted per definition, because there is no pre-segmentation and the consumer has to provide it himself). Another example is the idea of 'digest-viewing' [7, paragraph 5.1.4.2, (7)], in which the broadcaster might send summarisation-information (relevant segments) to be viewed for broadcasts of which the consumer missed the beginning.

12. Stream-switching
A more complicated version of 11 (but which may be experienced the same from a user's perspective) is stream-switching, in which the consumer is able to select camera-angles for example. Now there is not a single underlying programme, but there are multiple.

2.3 Other requirements

Besides implicit and explicit segmentation used by the consumer, there also can be a need for segmentation from other perspectives. As the consumer is totally unaware of the use of segmentation in this case, we have grouped them under 'other requirements'.

13. Seamless integration of already stored content.
When content is already available locally, it might be integrated with another broadcast. This may free-up resources for the recording of yet another programme, might allow cost-savings for the user (less NEW content was recorded), may allow extension of related broadcasts with already stored background content, etc.

14. Stream-switching
Stream-switching can also be necessary because of technical reasons at the broadcaster's side, like continuing a programme on another channel.

15. Updating of faulty broadcasts
(Technical) corrections of already broadcast material may be made unnoticed by the consumer. A piece of content that was broadcast with distorted audio, but has not been viewed yet may be updated automatically. The same goes for transmission-errors (maybe signalled by the STB), etc.
This might be difficult to achieve in practice, using MPEG-streams, because it asks for the transmission of parts of a programme only (only one audio track for example). We do address the implementation here, however.

16. Harvesting of consumer's viewing behaviour
The ability to do this up to frame-level might seem somewhat theoretic, but for (relatively short) 'programmes' like commercials the ability to registrate at which point the consumer stopped watching (or replayed the commercial) might be interesting for advertisers. More globally the amount of time (number of frames) spent on watching could be seen as an indicator for the commercial's attractiveness.

E-3 Absolute content reference

As was discussed in chapter 2, segmentation can be done at various levels. The requirement to be able to refer to content at frame-level was seen in multiple discussed services (editing, bookmarking, non-linear viewing). This implies the availability of an absolute content reference, like traditional time-code. However we propose a to use a slightly different approach from time-code.

The absolute reference:

1. Should be able to uniquely identify any piece of content up to the frame-level.
   Note that although the reference up to frame-level might be provided, the practical resolution of services might be limited because of GOP-structure for example.
2. Must enable identification of the current temporal position within a programme.
3. Must allow segmentation (from programme-level to frame-level).
4. Should survive broadcasting, multiplexing, recording, replay, file-storage, sharing, etc.
5. Should be easy to generate during production.
6. Plays an important role in the linkage of metadata to the content.
   Play-along games for example need synchronisation of the programme and other data.
7. Should be easily useable to build a 'viewingpath' (playlist).

3.1 USID

As a possible solution for the sketched absolute content reference, we propose the introduction of the USID*, which we will discuss now. The USID is an abbreviation for Uniform Segment IDentifier and extends the UGI-UPI concept. Basically it can be seen as an extension of traditional time-code to an absolute content-referencing mechanism including a link to the programme's UPI. It has to be noted here that although we will start the introduction with a specific definition of the USID, the details for how to compose and use it are not trivial. Therefore we will present a number of issues which may be useful as discussion starting-points.

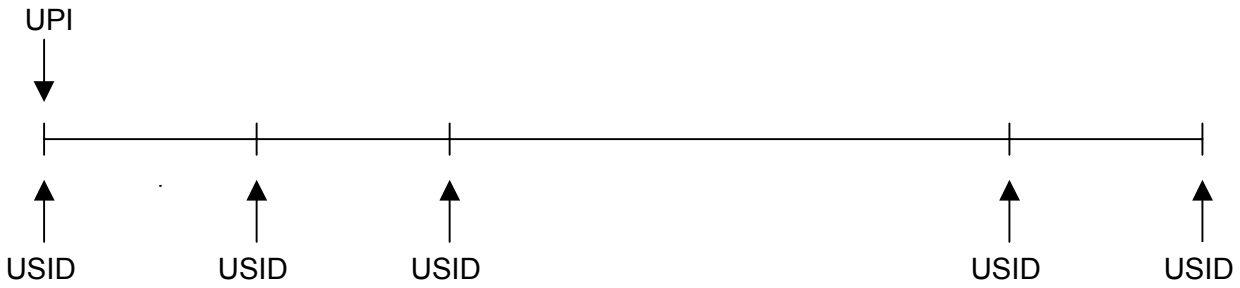


Figure 1 Programme containing USIDs.

Figure 1 shows the time-line of a programme referred to by a UPI and segmented by five USIDs. At first glance the USIDs may be seen as some form of time-code, because every frame could have its own USID (the ones drawn might be the only ones referred to). The USID is more sophisticated, however. Figure 2 sketches its syntax. Figure 3 gives the syntax of the USID's relative, the extended UMID. It is provided here to allow easy comparison of both identifiers. We will discuss the relation to the UMID later.

** Within EBU the 'USID' is referred to as 'Unique Source IDentifier'. The choice of another name might thus be required to prevent unintended confusion.*

unilabel	L	Version number	Mapped UPI	Time & Date (SMPTE)	Extension field
12	1	3	16	8	0 - 228

Figure 2 USID syntax.

unilabel	L	Inst. number	Mat. #	Time & Date (SMPTE)	Spatial	Country	Org	User
12	1	3	16	8	12	4	4	4

Figure 3 Extended UMID syntax.

Let us now define what the different fields indicate for the USID (for the UMID the definitions can be found in [5]).

Unilabel: Identifies USID as a SMPTE USID. If the USID-concept is used SMPTE will provide us with an appropriate value for this label. The unilabel is defined in [8].

L: Length, the length of the remaining parts of the USID, used for Key-Length-Value coding. If all values are allowed, the maximum (255) will limit the extension field to 228 bytes max.

Version number: Identifies the version of the programme. This field will be explained in more detail below.

Mapped UPI: Identifies the programme this USID belongs to. This field will be explained in more detail below.

Time & Date: This field (together with the unilabel, version number and Mapped UPI) makes the USID unique and allows it to be used as a frame-accurate reference. The time/date syntax is defined in SMPTE 309M.

Extension field: This field may be used for various purposes. It will be discussed below.

3.1.1 Version number

The version-number field is meant to identify different versions of the same programme. Although it was remarked on the myTV-reflector that different programme-versions should have different UPI's, this might not be the case for programme(-segments) rebroadcast because of technical failures for example (also see paragraph 2.3, 15). This field allows for those versions to be identified. The field's structure is sketched in figure 4.

Userflag	Version
1b	2 + 7b

Figure 4 Version-number syntax.

Userflag: This field identifies the version number as a consumer-version number. In this case the Mapped UPI field is interpreted as a user-ID. This field shall be 0 for original (broadcast) material.

Version: The version-number identifies different versions of the same material. Although this field may not be needed for broadcast material (depending on the definition of the use of the UPI), it is needed for consumer-generated material (editing, capturing, etc.), because it is the only way to differentiate different programmes generated by the same consumer. Note that this is because the Mapped UPI will have a constant value for consumer-material. The maximum number of consumer-generated content is: 8388608 different 'programmes'.

3.1.2 Mapped UPI

The Mapped UPI field serves as a unique material identifier. Because we already possess a unique material identifier, the UPI, a mapping of UPI to this number seems a simple way to generate it. A problem might be that the UPI (as far as the authors understand) is not limited in length, while the Mapped UPI field has 16 bytes. 16 bytes can generate $3{,}4^{E}38$ UPI's. When we assume the UPI to only use 26 different characters this is the equivalent of approximately 27 characters in a UPI. This may lead to problems for the mapping. One simple mapping would be to use the base URL to map the first part and then count programmes on top of this. But how many characters should be used for the base URL? More importantly an URL might have too much characters to use as a base URL and thus the mapping is not clear then. The 'counting' would require a broadcaster to keep track of the programme-number. This might be a drawback, but on the other hand generation of unique UPI's also asks for some way of knowing what was broadcast in the past (also see BBC R&D Technical Note 1407(00) on the subject of trailer to programme-linking).

A better alternative might be to use the Unique Material ID (UMID) as it was generated by the last editing/copying equipment, to fill the Mapped UPI field. This is very convenient, because if UMIDs are used, they allow easy generation of USIDs from the material-number. The question now is, do we want the USID to change with the material, just as the material-number does (for example at shot-level)? This would provide a way to find back the original material somehow, but it seems not necessary to have this detailed information in the home. Also if the same programme is copied for broadcast by another broadcaster; is the UMID changed? (The instance-number might be, but the material-number should not be for our scheme to work.)

For consumer-generated material the Mapped UPI field may be generated locally. An example of this is capturing home-shot material, or re-editing broadcast material. When content is 'really new' a consumer Mapped UPI may be flagged in the version-field and the version-counter (which should be persistent for the TV-Anytime device in the home) be increased. In this case the Mapped UPI field should contain a unique User ID. May be this can be filled in as an IP-address, which is also 128 bits long (IPv6). If the box has a unique address this might be a simple but effective way to generate unique consumer ID's without additional registration. One drawback is that multiple users are not accommodated automatically, but as the version-number allows for more than 8 million programmes, this does not seem a problem (the number-space can of course be divided among multiple users).

3.1.3 Time & date field

The Time & date field is meant to provide SMPTE-time code with date-support. Thus preventing loop-around problems on (very) long recordings and allowing for the easy comparison of the field's values. It thus facilitates easy uniqueness for all material.

The value of this field is meant to be generated in the final authoring process, because it should stay the same if the same programme arrives at the box in different ways (for example broadcast and Internet). Also it seems logical to demand incremental values only (no doubles), just as with traditional time-codes. Jumps may be allowed, however.

These requirements can facilitate easy navigation when only start and end-USID are known. The player then just has to play from the start TC (or the first value higher than it in case of incomplete material) and stop before the end-TC (or up to and including the first value lower than it in case of incomplete material).

3.1.4 Extension field

The use of the extension field is optional and allows extension of the USID in various ways. The extension is signalled by the L-field. One example of the extension could be to define a segmentation flag, which allows segments to be identified 'on the fly' or even later generated by editing this bit. A set flag may indicate this segment's USID should be added to a local table of segments providing entry-points to the programme (examples of those tables are discussed in paragraph 4.3).

Another way of achieving this goal is through the use of segmentation table signalling. Adding a segment then would mean sending an altered table. The latter seems preferable, because it doesn't involve editing of the material itself. This decoupling of content and its segmentation makes it more consistent with later or local segmentation.

Another use for the extension field might be to incorporate object-links to metadata-information (SMPTE).

3.2 Relation to UMID

The USID is based on the notion of a UMID, as proposed by SMPTE in [5]. The UMID is assumed to be locally created in the content-generation & authoring process and can easily be adapted for use in TV-Anytime services (compare figures 2 and 3). This characteristic is seen as a major benefit in coming to a standardized way of referring to content. Also the use of sets and packs and efficient KLV-coding techniques are 'inherited' in this way (see SMPTE-proposals for specific information).

3.3 Implementation-issues (informative)

On the conceptual level, we feel an absolute content-reference for every frame should be provided. This can be seen as a 'high-level' approach which doesn't want to be concerned about underlying stream-structures.

We do not discuss how to implement the USID. It could for example be implemented by sending a USID-header in some way and only incrementing the time & date field for the next frames. The latter might be done using PTS/DTS time-stamps within MPEG streams. An important aspect with MPEG, is the underlying GOP-structure. If playing (and later editing) may only be done at I-frames, is there any use to be able to refer to all frames? Maybe sending a USID at I-frames only would be enough in that case?

A main difference with our USID-concept and current standards suitable for implementation*, is that the USID is seen as an absolute reference that is closely linked to the content and not meant for altering 'down the line'. It could be seen as a sophisticated WORM-TC facilitating unambiguous content-referencing. Also the USID is assumed to be generated in the final content-authoring stage and (typically) not in the final broadcast-encoder.

When using PTS/DTS time-stamps for example, splicing-devices 'down the line' (like local add-inserters) must make sure the references to PTS/DTS time-stamps are updated whenever the PTS/DTS-information is re-stamped by those devices. This is a very important requirement when implementing absolute references using PTS/DTS (or any other 'restampable') timing mechanism and is a main difference with the USID-concept (but might still be a practical implementation of it). Another difference with the USID is that the latter is based on time & date and doesn't wrap around once per day.

** Related issues on content referencing are addressed in DSM-CC (which also facilitates jumping to certain points in the content) and SMPTE 312M (Splice Points for MPEG-2 Transport Streams) for example.*

E-4 Segmentation & metadata tables

This paragraph discusses some simple mechanisms to facilitate services using the USID introduced earlier. We will introduce the concept of SEGMET-tables to add metadata to segments and we will discuss viewingpaths and related functionality introducing VIEWPATH-tables. Note that for these mechanisms to work, we assume the USID is provided for every frame.

4.1 SEGMET-tables

A simple way to provide additional metadata at segment-level is the use of SEGMET-tables (SEGment METadata). Such a table consists of USID's with related metadata, as is sketched in figure 5.

Start USID	End USID	Metadata
USID0	-	Start of programme
USID223	USID2333	Parental guiding: 18+ only
USID3434	-	End of programme

Figure 5 Table illustrating the SEGMET-concept.

The sketched table facilitates the specification of start of segments and end of segments (optional). This allows two USIDs to be used to identify a complete segment within the programme. Note that if the time & date field in the USID is incremental only (may be with positive jumps), even when frames are missed the SEGMET-table still links the correct metadata to the correct segments.

SEGMET-tables can also be used to indicate grouping of segments, as is indicated in figure 6.

Start USID	End USID	Metadata
USID0	USID2122	Gamegroup
USID2122	USID4500	Break
USID4501	USID9993	Gamegroup

Figure 6 SEGMET-table used to group segments in two groups.

Ones such a grouping is created (and locally stored), it might be referred to in content-searches or 'viewing from a menu' (the consumer can pick the segment he would like to watch). The latter is illustrated in figure 7. This SEGMET-table shows the explicit identification of segments as chapters, allowed to be jumped to by consumers.

Start USID	End USID	Metadata
USID0	USID2122	Chapter1 'First round'
USID2122	USID4500	Chapter2 'Break'
USID4501	USID9993	Chapter3 'Second round'

Figure 7 SEGMET-table used to identify entry-points for the consumer explicitly.

4.2 Means of delivery

Just as with the USID we do not specify how to deliver the SEGMET-tables. Again a lot of related standards seem to exist already (think of DVB-SI, etc.). These could provide the practical implementation to the SEGMET-data delivery. Requirements may be imposed on the speed (or frequency) of which the tables are provided.

For example: to identify a goal in a live soccer-programme, rapid broadcast of a SEGMET-table might be demanded. The table will always be send after the goal, however. Note that it depends on the application if complete SEGMET-tables have to be delivered. For a large amount of applications we feel this may not be necessary (then the SEGMET-table can be seen just as an 'extender' of the data-capacity of the original programme).

4.3 Viewingpaths

Viewingpaths basically are playlists. We use a different term, to avoid confusing with traditional 'play-out'-playlists and to take into account that the lists may very well be generated by the consumer's viewing-behaviour (explicitly or implicitly asked for by selecting a specific story-line). We will now introduce the viewing-path table and refer to it as the VIEWPATH-table.
The most simple viewing path would be to view a programme linearly, for which a table is given in figure 8.

Start USID	End USID	Options
USID0	USID999990	-

Figure 8 Simple VIEWPATH-table.

A bit more complex VIEWPATH-table is shown in figure 9. Here the options-field is used to identify a piece of content to be shown in slow-motion. The USIDs are referred to with letters now also, to indicate they refer to different programmes (normally this distinguish of course is part of the USID's value). The application might be a movie with two story lines (from two separately broadcast programmes) containing a flashback (in slow-motion).

Start USID	End USID	Options
USIDA0	USIDA93	-
USIDB33	USIDB3434	-
USIDA45	USIDA93	Slowmo
USIDB3434	USIDB99029	-

Figure 9 VIEWPATH-table also making use of the options-field.

Figure 10 gives a VIEWPATH-table that provides information on three different story-lines that may be followed by the consumer. Note that the grouping could be signalled in the SEGMET-tables also, but the ordering could not (except when a group + number was used in the SEGMET-table, of course*.).

** Actually this introduces the question of what information should be where. We started out with the basic metadata/viewingpath difference, because it is intuitive. It could of course very-well be sub-optimal. We see it is a conceptual simple enough starting point for further discussion.*

Start USID	End USID	Options
USIDA0	USIDA93	Story1
USIDB33	USIDB3434	Story1
USIDA45	USIDA93	Story2, Slowmo
USIDC0	USIDC23	Story2
USIDC933	USIDC34934	Story2
USIDC23	USIDC3344	Story2
USIDC5000	USIDC55555	Story3
USIDA1000	USIDA2000	Story3

Figure 10 VIEWPATH-table introducing three story-lines

The consumer might build his own VIEWPATH-table, during viewing (implicitly or explicitly). The box might registrate the following events:

1. Store USID + viewing options where viewing has started**.
2. Check continuous incrementing of USID, if not; store 'missed USID's + viewing options'.
   *This is necessary to be able to provide exactly the same viewing experience when replayed/shared, etc. Because the 'missed frames' might be recorded later for example.****
3. Store USID where viewing for this programme has stopped (if switched to another programme for example).
4. Store USID + viewing options where viewing continuous (this might be a jump within or to another programme).
5. Store USID + options where viewing-options have changed (trick-play, etc.)

And also:

6. Store consumer's comments, such as: 'blocked for my children', commercial-rating, etc.

*** Of course only 'non-normal' options may be stored.*
**** It the reason for missing frames is known this could of course be done more gracefully.*

E-5 Discussion topics

As was remarked already, this proposal is a starting document only. It originates from questions within the myTV-project. We have tried to identify important discussion-topics, which have not been concluded on within myTV. Opinions thus should be seen as current opinions of the authors and do not necessarily conform to the final ones in the myTV group. This chapter gives some more issues for discussion.

p 3.1.1 The user-flag influences the Mapped UPI field, this may be less 'elegant'.

p 3.1.1 When exactly is a new UPI created? For example: do we need a new UPI or version-number for updated AA-news-segments?.

p 3.1.2 Home-edited programmes from broadcast material may have copyrights on them, if they should stay with the original material changing of its USIDs may be unwanted. On the other hand having a continuous time & date-code within the USID for re-edited material may be easier to manage than having large 'jump-tables'. What route to take?

Other: What about other types of material? Audio and video data may require different USIDs*.

* SMPTE 330 M also defines audio time-stamps. Extending the USID for audio-data (and possibly any data) seems necessary to be able to use audio-fragments (samples for play-along games, etc.) and for future consumer-editing, home audio-recording, etc.

E-Appendix A Related requirements in TV037r2

This appendix provides an overview of related requirements in [3]. It is meant to facilitate discussion on this proposal in relation to the metadata requirements. Comments are included in brackets.

Par.	Number	Short descripiton
5.1	1	*Allow user to find, access, navigate content.*
5.1	2	*Based on open standard (SMPTE/TV-Anytime).*
5.1	3	*Evolution of metadata specification (SMPTE-dictionary element).*
5.1	4	*Upgradeable in a flexible manner (Create new type in dictionary?).*
5.2	5	*Interoperable with TV-Anytime content referencing system (mapped UPI).*
5.2	8	*Interoperability through common core set of metadata.*
5.2	10	*Designed for interoperability between different sources, regardless of distribution. (Universally unique absolute content reference)*
5.2	11	*Mechanisms to interoperate with existing metadata standards (SMPTE).*
5.4	15	*Bandwidth efficient (KLV-coding).*
5.5	20	*Forbid content or segment skipping (through SEGMET-tables).*
5.5	21	*Parental guiding (through SEGMET-tables).*
5.6	25	*Synchronisation between content and metadata (frame-accurate with USIDs).*
5.6	26	*Capture and playback real-time/non real-time (TC increments only).*
5.6	30	*Synchronize metadata and content (through USID, optionally extension field).*
5.6	31	*Recording of segments only provides access to metadata (table-frequency).*
5.6	32	*On-the-fly creation of metadata (tables, optionally flagging in USID).*
5.6	33	*Ignore irrelevant metadata (through KLV-filters).*
5.6	36	*Custom metadata (private sections in SMPTE dictionary?).*
5.6	38	*Synchronized metadata at different granularities (ranges of USIDs).*
5.6	39	*Non-linear viewing and trick-modes (signalled in SEGMET, use USIDs to play).*
5.6	40	*Adding personal metadata and bookmarking (create SEGMET-tables locally).*
5.6	41	*Synchronisation with applications/stored content (frame-accurate USIDs).*
5.7	48	*Multi-lingual data and metadata keys (supported by SMPTE dictionary).*
5.7	50	*Different views of a programme (VIEWPATHS).*
5.7	51	*Multiple instances of the same content (version number).*
5.7	53	*Support commerce models like PPV (detailed consumption records via USIDs).*
5.7	54	*Collection of viewing history information (record USIDs/ranges that were viewed).*

Appendix F : AN166

Included below is the text of a proposal/contribution to the TV-Anytime Forum, jointly made by the myTV project partners, in September 2000. It deals with content referencing over IP.

Introduction

This document describes the current thinking from the members of the myTV project on how location resolution will be carried out over an Internet link within the myTV project. It is submitted to the TV-Anytime forum as part of the call for contributions on location resolution over a bi-directional network.

Chosen solution

It was felt that within the myTV project we wanted to keep the software requirements on the PDR as low as possible, which meant that we wanted to re-use an existing Internet protocol for location resolution. It was decided that a HTTP v1.0 protocol was sufficient for our needs whilst being reasonably easy to implement.
The request to resolve a CRID is sent to the remote server using a HTTP request using the standardized rules for making a CGI request. Obviously it is the option of the location resolution service provider to implement this service using any server side technology they wish (CGI scripts, java servlets etc).
The data returned by the server could be in any format we desire, but again we felt that it was better to re-use existing technology. As metadata is likely to be accessed using some form of XML (maybe using XML schemas or XML + DTD) it was decided that using XML for the response from the server was desirable.

Forming the Query String

A mandatory parameter to the HTTP request is the *CRID*. The syntax for the HTTP request is:

http:*//*<*location of server*>**?CRID=**<*crid*>

Where <*location of server*> is the location of the location resolution server given in the resolution authority record for the CRID's authority. <*crid*> is the CRID to resolve.
A trivial extension allows the PDR to resolve multiple CRIDs using a single request, thus minimising the number of HTTP requests that have to be made.

http:*//*<*location of server*>**?CRID=**<*crid1*>;***CRID=***<*crid2*>;***CRID=***<*crid2*> … ;***CRID=***<*cridn*>

This could give rise to a URL such as,

```
http://www.bn.com/myTVservices.cgi?CRID=crid://bn.com;abc;CRID=crid://bn.com;def
```

Format of the XML Response

```
<!ELEMENT LOCATION_RESOLUTION_TABLE (PROGLOC)+>
<!ATTLIST LOCATION_RESOLUTION_TABLE VERSION CDATA #REQUIRED>
  <!ELEMENT PROGLOC (CRID* | URL* | RESOLUTION_AUTHORITY*)>
    <!ATTLIST PROGLOC
              QUERY_CRID    CDATA                          #REQUIRED
              COMPLETE      (TRUE | FALSE)                 "TRUE"
              TYPE          (COLLECTION | ALTERNATIVE)     "ALTERNATIVE"
              STATUS        (0 | 1 | 2 | 3)                #REQUIRED>
    <!-- The semantics of the status attribute are as follows:
         1) Invalid CRID,
         2) Cannot yet be resolved - try again (CRID is recognized
            and has been postponed, or is not yet resolvable),
         3) Unable to resolve. The meaning of this depends on
            whether it is a primary or secondary authority. -->

      <!ELEMENT CRID                  (#PCDATA)>
      <!ELEMENT URL                   (#PCDATA)>
      <!ELEMENT RESOLUTION_AUTHORITY (#PCDATA)>
```

Figure 1: The DTD for a location resolution response. Note, in particular, the exact meaning of each of the status codes.

The *STATUS* attribute can take one of four values, whose meaning is given in the DTD in Figure 1. (An example of use is shown in Figure 2.) Numeric values are used since these are easier for the PDR to interpret, and can be easily extended. Using the value, along with knowledge of whether the resolving authority is primary or secondary, tells the PDR whether or not the CRID should be discarded or another resolution attempt should be made later. Table 1 specifies the action which a PDR should take for the different scenarios.

Error Type	Primary Authority	Secondary Authority
0 Success	Resolution successful - store CRID and children[1]	Resolution successful - store CRID and children
1 Invalid CRID	Discard CRID	Discard CRID
2 Cannot yet resolve	Try again later	Try again later
3 Unable to resolve	Discard CRID	Try other authorities (if available), otherwise try again

Table 1: How the STATUS attribute affects actions taken by a PDR on receiving a resolution response.

The format of the response depends on the value of the *STATUS* attribute. If this is "0" (success) then each `PROGLOC` element can contain one of three responses.

1. A list of one or more CRIDs. (Query CRID is a group CRID.)
2. A list of one or more locator URLs. (Query CRID is a leaf CRID.)
3. A list of one or more resolution authorities. (Query CRID was assigned by content creator, not content provider.)

The input to the location resolution server may or may not be a group CRID. The PDR is able to determine from the response whether the original CRID is a group or not (regardless of whether we chose to indicate this in the naming of the CRID). This information is implicit, since we know that if a CRID list is returned, the original CRID was a group CRID.

[1] Depending on the nature of the CRID's children, and the value of the *COMPLETE* attribute, further resolutions may be required.

Examples of XML response

The first two types of responses listed above will be the most common and some examples of these are given in Figure 2 and Figure 3. In particular, note the different ways in which the *COMPLETE* and *TYPE* attribute are combined and what this means.

```
<?xml version = "1.0"?>
<!DOCTYPE LOCATION_RESOLUTION_TABLE SYSTEM "location_resolution_list.dtd">
<LOCATION_RESOLUTION_TABLE VERSION="1.0">
  <!-- This is an example of a single program being broadcast on DVB, which is
then repeated 4 days later. There is no need to resolve again (so
COMPLETE="TRUE") and the PDR should only capture one instance of the children
(so TYPE="ALTERNATIVE"). -->
  <PROGLOC QUERY_CRID="crid://bbc.co.uk;11550"
           COMPLETE="TRUE" TYPE="ALTERNATIVE"
           STATUS="0">
    <URL>dvb://bbc1.1a63.09@19981124T1845D0045</URL>
    <URL>dvb://bbc1.1a63.09@19981128T0930D0045</URL>
  </PROGLOC>
  <!-- Here, the original CRID refers to a series which resolves into CRIDs
referring to each episode. These CRIDs could be resolved, in turn, as above. The
series is ongoing so the PDR is instructed that the list is not complete. -->
  <PROGLOC QUERY_CRID="crid://bbc.co.uk;13174"
           COMPLETE="FALSE" TYPE="COLLECTION"
           STATUS="0">
    <CRID>crid://bbc.co.uk;13174.03</CRID>
    <CRID>crid://bbc.co.uk;13174.04</CRID>
    <CRID>crid://bbc.co.uk;13174.05</CRID>
  </PROGLOC>
  <!-- In this case, the CRID is invalid so an error message is returned. Note,
that when there is an error (or a resolution authority list is returned) the
other attributes of PROGLOC are meaningless so are not included. -->
  <PROGLOC QUERY_CRID="crid:/bbc.co.uk;1374" STATUS="1"/>
</LOCATION_RESOLUTION_TABLE>
```

Figure 2: A location resolution response demonstrating the resolution of a leaf CRID, a group CRID and an incorrect CRID

The *COMPLETE* attribute indicates to the PDR whether it should attempt to resolve this CRID again (assuming a successful response). For example, a three episode drama might have a 'TRUE' value, whilst an ongoing soap opera would have a 'FALSE' value. Although this indicates complete resolution of the parent (i.e. query) CRID, this does not indicate that children nodes will not need further resolution.

The *TYPE* attribute is used by the PDR to determine how the returned list of CRIDs or locator URLs should be handled. If the value is 'COLLECTION' then the PDR will attempt to capture all the elements in the list. If the value is 'ALTERNATIVE' only one of the elements will be captured. Figure 3 gives two different examples of how these attributes might be combined.

```
<?xml version = "1.0"?>
<!DOCTYPE LOCATION_RESOLUTION_TABLE SYSTEM "location_resolution_list.dtd">
<LOCATION_RESOLUTION_TABLE VERSION="1.0">
  <!-- This is an example of a single program being broadcast on DVB then
repeated 8 days later on another channel. Another repeat is planned but the
schedule is not yet known, so the request is incomplete. (This is the most
unlikely combination of the COMPLETE and TYPE attributes.) -->
  <PROGLOC QUERY_CRID="crid://rtvslo.si;11550"
           COMPLETE="FALSE" TYPE="ALTERNATIVE" STATUS="0">
    <URL>dvb://rtvslo..slo1;1a639f@19981116T1845D0045</URL>
    <URL>dvb://rtvslo..slo2;1a639f@19981124T1845D0045</URL>
  </PROGLOC>
  <!-- Here, the original CRID refers to a drama which resolves into 4 CRIDs
referring to each episode. There are a known and fixed number of episodes so the
request is complete. -->
  <PROGLOC QUERY_CRID="crid://rtvslo.si;abcd"
```

```
            COMPLETE="TRUE" TYPE="COLLECTION" STATUS="0">
    <CRID>crid://rtvslo.si;abcd_01</CRID>
    <CRID>crid://rtvslo.si;abcd_02</CRID>
    <CRID>crid://rtvslo.si;abcd_03</CRID>
    <CRID>crid://rtvslo.si;abcd_04</CRID>
  </PROGLOC>
</LOCATION_RESOLUTION_TABLE>
```

Figure 3: Another location resolution response, this time demonstrating a different use of the *TYPE* and *COMPLETE* attributes.

Finally, we need to consider the third type of response, as shown in Figure 4.

```
<?xml version = "1.0"?>
<!DOCTYPE LOCATION_RESOLUTION_TABLE SYSTEM "location_resolution_list.dtd">
<LOCATION_RESOLUTION_TABLE VERSION="1.0">
<!-- The query cannot be directly answered by the resolution authority (who is
the content creator) so a list of resolution authorities is returned. -->
  <PROGLOC QUERY_CRID="crid://warner.com;Friends" STATUS="0">
    <RESOLUTION_AUTHORITY>nbc.com</RESOLUTION_AUTHORITY>
    <RESOLUTION_AUTHORITY>sky.com</RESOLUTION_AUTHORITY>
    <RESOLUTION_AUTHORITY>channel4.com</RESOLUTION_AUTHORITY>
    <RESOLUTION_AUTHORITY>rtl.co.de</RESOLUTION_AUTHORITY>
  </PROGLOC>
</LOCATION_RESOLUTION_TABLE>
```

Figure 4: An example of a location resolution process, which leads to a list of resolution authority names.

This type of response is used to resolve CRIDs issued by the content creator. From the response the PDR can then attempt to resolve the CRID using the resolving authorities listed. The PDR needs to choose which resolving authority in the list to use first. The obvious approach is to first try those resolving authorities for which the PDR has cached the Resolving Authority Record. Amongst these, the resolving authority that has most recently been used should be tried first.