The *TV-Anytime* Forum

Specification Series: S-3

On:

Metadata

(Normative)

NOTICE

Use of the technologies described in this specification may infringe upon patents, copyrights or intellectual property rights of Members or non-Members of the *TV-Anytime* Forum.

Although the *TV-Anytime* Forum makes a reasonable effort to ascertain the ownership of intellectual property as an aid to the users of its specifications, it is entirely the responsibility of individual users of the specification to obtain all necessary licenses.

Neither the *TV-Anytime* Forum nor any of its Members accept any responsibility whatsoever for damages or liability, direct or consequential, which may result from use of this specification.

Document: SP003v1.1
Date: August 17, 2001

Series Overview

This is the third in a series of five “S-series” documents produced by the *TV-Anytime* Forum. These documents establish the fundamental specifications for the services, systems and devices that will conform to the *TV-Anytime* standard, to a level of detail that is implementable for compliant products and services.

As is common practice in such standardization efforts, these specification documents were preceded by requirements documents (“R-series”), which define the requirements for the *TV-Anytime* services, systems, and devices.

Congruent with the structure defined in *TV-Anytime*’s Call for Contributions (TV014r3), these specifications are parsed into three major areas, each described in a separate document of the series: Metadata (S-3), Content Referencing (S-4) and Rights Management (S-5). See the Call for Contributions for more detail on the derivation and background of these categories and their respective roles in the *TV-Anytime* standardization process.

The other two documents in the S-series are intended to define the environment and system architecture in which the standards in S-3, S-4, and S-5 are to be implemented. The first document in the series (S-1) provides benchmark business models against which the *TV-Anytime* system architecture is evaluated to ensure that the *TV-Anytime* standard enables key business applications. The next document in the series (S-2) presents the *TV-Anytime* System Architecture. These two documents are placed ahead of the other three for their obvious introductory value; S-1 and S-2 are both informative.

Although each of the S-series documents is intended to stand alone, a complete and coherent sense of the *TV-Anytime* system standard can be gathered by reading all five of the specification documents in numerical order.

Metadata Specification S-3
Document Revision History

Change History	
16 Feb 2001	Initial release Version 1.0
14 Apr 2001	Released corrections to V1.0 as V1.0R1
17 Aug 2001	Initial release Version 1.1

Table of Contents

Figures

About the TV-Anytime Forum

The global *TV-Anytime* Forum is an association of organizations which seeks to develop specifications to enable audio-visual and other services based on mass-market high volume digital storage in consumer platforms – simply referred to as *local storage*.

The *TV-Anytime* Forum was formed at an inaugural meeting held in Newport Beach, California, USA, on 27-29 September 1999. It has started work to develop open specifications designed to allow Consumer Electronics Manufacturers, Content Creators, Telcos, Broadcasters and Service Providers to exploit local storage.

As part of its formation, the *TV-Anytime* Forum has established four fundamental objectives for the organization, which are:

- The *TV-Anytime* Forum will define specifications that will enable applications to exploit local persistent storage in consumer electronics platforms.
- The *TV-Anytime* Forum is network independent with regard to the means for content delivery to consumer electronics equipment, including various delivery mechanisms (e.g. ATSC, DVB, DBS and others) and the Internet and enhanced TV.
- The *TV-Anytime* Forum will develop specifications for inter-operable and integrated systems, from content creators/providers, through service providers, to the consumers.
- The *TV-Anytime* Forum will specify the necessary security structures to protect the interests of all parties involved.

Member organizations from Europe, the USA, and Asia, are drawn from a wide variety of industries: Traditional Broadcasters, Internet Broadcasters, Content Owners, Service Providers, Telcos, Consumer Electronics Manufacturers, IT Industries, Professional Equipment Manufacturers, Component Manufacturers and Software Vendors.

The *TV-Anytime* Forum invites *participation* from all interested organizations. Membership is open to all who sign the Memorandum of Understanding (published as document TV0004) and attend meetings. Meetings are held approximately every two months in Europe, the USA, and Asia.

For more information or to get involved with the work of the *TV-Anytime* Forum, visit the *TV-Anytime* Forum (www.tv-anytime.org) or contact:

Chairperson:	Simon Parnall (sparnall@ndsuk.com)
Vice-chairpersons:	Henry D. Chadwick (hdchad@us.ibm.com)
	Wataru Kameyame (wataru@mn.waseda.ac.jp)
Administrator:	Lynne Apodaca (lapodaca@inventures.com)
	(Tel: +1.925.275.6653)

1. Scope

We use the term "metadata" to mean descriptive data about content, such as program title and synopsis. We call such metadata "attractors" because they can attract a consumer to content. Attractors allow consumers to find, navigate and manage content from various sources. In addition to attractors, metadata as defined by TV-Anytime also includes information about user preferences and history. User preference information, such as favorite actors or TV shows, is included within the scope of TV-Anytime metadata to allow software agents to select content on the consumer's behalf.

The set of metadata described in this document was selected in order to satisfy the usage scenarios listed in the TV-Anytime business models requirements document R-1. The formal definitions of metadata schemas should be read in conjunction with the system specification defining how they could be used in an end-to-end system

TV-Anytime only defines the metadata format for metadata that may be exchanged between various entities such as between the content provider and consumer, among consumers, or between a third-party metadata provider and the consumer.

The metadata “representation format” defined here is the formal specification of how TV-Anytime metadata is represented in XML. Although XML Schema is used to define how metadata is represented in XML, it can also be used to describe equivalent, non-XML representations of the same metadata. For example, TV-Anytime metadata could be encoded in a binary format for transmission or storage.

This document defines the metadata schemas that are used within the overall TV-Anytime system. The TV-Anytime Systems Specification SP002v1.1 defines how these schemas are used in an end-to-end system. Note that the transport mechanism for metadata is out of scope of TV-Anytime; it must be specified by other bodies such as DVB, ATSC and ARIB. Furthermore, the manner in which metadata is stored, accessed and used on the PDR is also out of scope of this specification.

2. Glossary of Terms

Attractor	A metadata element that is accessible by the consumer in order to aid in the content selection process, thus attracting the consumer. Examples include the title and name of an actor in a television program
Application	A specific set of functions running on the PDR. Some applications use metadata, either automatically or under *consumer* control
Content Creator	The producers of the content
Content Reference	A pointer to a specific content item
Content Provider	An entity that acts as the agent for and is the prime exploiter of the content
Description Scheme	A formal definition of a metadata schema written in the MPEG-7 Description Definition Language [3].
Descriptor	A metadata element, such as an attractor or other information about content such as the key frame index of a piece of video
Enhanced TV	Television that includes additional information and/or applications related to content, but does not use a return path
Interactive TV	Television that includes additional information and/or applications related to content and which takes advantage of a return path
Life Cycle	The process of creation, usage, storage, and deletion of metadata
Location Resolution	The process of establishing the address (location and time) of a specific content instance from its CRID
Namespace	An identifier associated with a set of XML schemas that globally identifies those schemas so that they can be referenced externally. A globally unique namespace ensures that the names of types defined by schemas in that namespace do not conflict with types of the same name defined elsewhere.
Metadata	Generally, data about content, such as the title, genre and summary of a television program. In the context of *TV-Anytime*, metadata also includes consumer profile and history data.
Metadata Schema	A set of rules describing the syntax and semantics of metadata
Metadata System	The collection of components that allows the end-to-end operation of the *TV-Anytime* metadata solution
Omnibus Program	A single program that consists of a sequence of related programs, for example, the week's episodes of a daily TV serial.
Program	An editorially coherent piece of content. Typically, a program is acquired by the PDR as a whole.
Program Group	One or more programs that are grouped together. TV-Anytime defines several types of program groups such as "series" and "programme compilation".
Return Path	Part of the bi-directional distribution system from the consumer to service provider
Segment	A continuous portion of a piece of content, for example a single news topic in a news program.
Segmentation	The process of creating segments from a piece of content.

2.1 Abbreviations

ATSC-DASE	A set of application programming interfaces currently being standardized by the Advanced Television Systems Committee for the digital broadcast of multimedia and applications in North America and other regions
CFC	Call for Contributions
CRID	Content Reference IDentifier, an identifier for content that is independent of its location
DDL	Description Definition Language, the language used to define description schemes in MPEG-7. See [3].
DVB-MHP	A set of application programming interfaces being standardized by Digital Video Broadcasting Project for the digital broadcast of multimedia and applications in Europe, Asia, and other regions
EBU p/meta	A metadata format specified by the European Broadcasting Union
EPG	Electronic Program Guide: A means of presenting available content to the consumer, allowing selection of desired content
HTML	Hypertext Markup Language
IPR	Intellectual Property Rights
MPEG-7	Ongoing effort by the Motion Pictures Expert Group to specify a standard set of content-related metadata applicable to a broad range of applications
PDR	Personal Digital Recorder
SMPTE	Society of Motion Picture & Television Engineers. Also a standard for video production metadata.
XML	Extensible Markup Language

3. Introduction

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

The *TV-Anytime* metadata system allows the consumer to find, navigate and manage content from a variety of internal and external sources including, for example, enhanced broadcast, interactive TV, Internet and local storage. It defines a standard way to describe consumer profiles including search preferences to facilitate automatic filtering and acquisition of content by agents on behalf of the consumer. Consumers, as used in this document, include educators and students, who may use selected program segments in the classroom or laboratory.

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

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

3.1 TV-Anytime Process Model

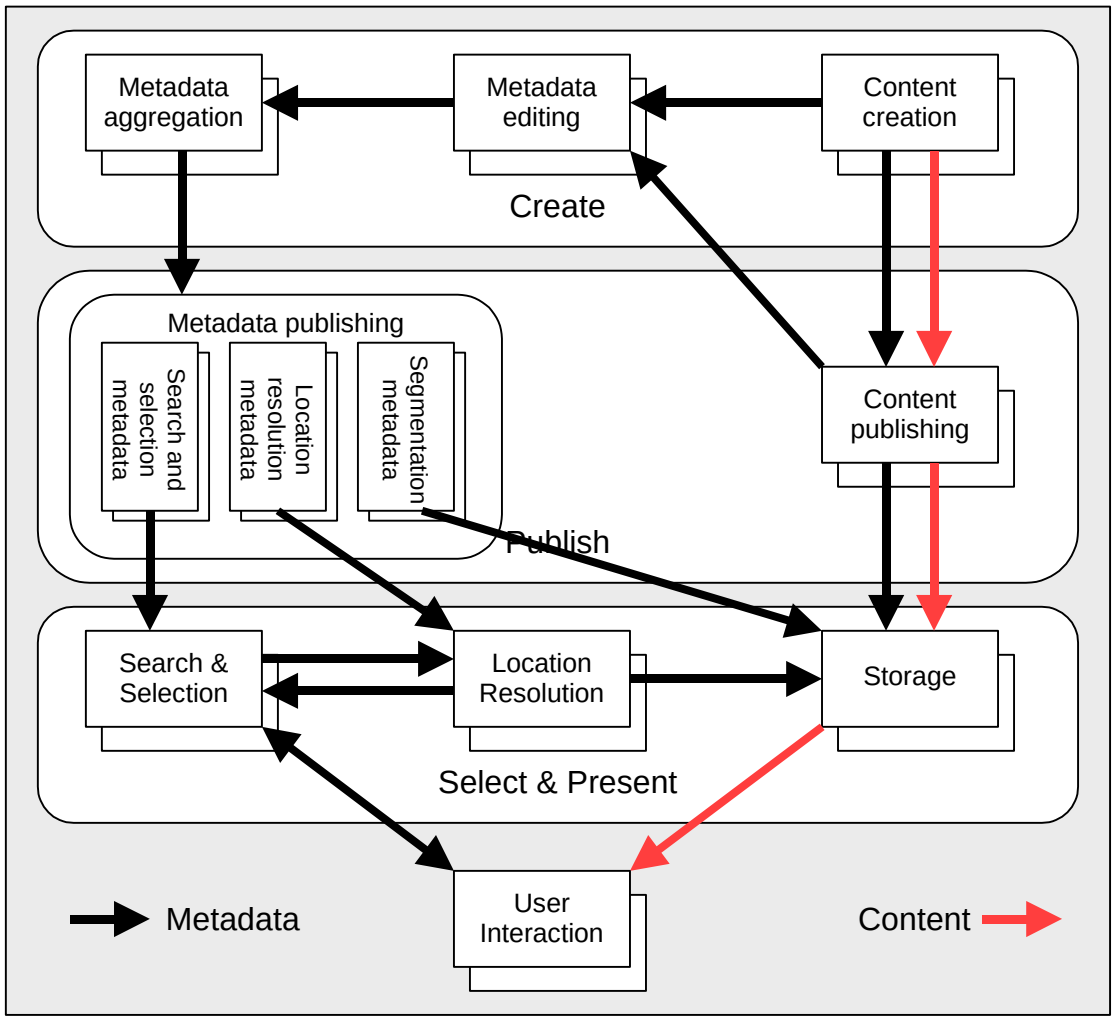

Figure 1 Metadata and Content flow

Figure 1 shows the flow of metadata and content through various stages of creation and delivery to the end consumer.

This model clearly identifies the separation of the processing of metadata and content while at the same time illustrating the parallels between the processing of metadata and content. User profile and history metadata is generated during the selection and presentation process.

Content creation

The content creation process represents the production of a piece of content or a program. During the production process, the program content is created and information about the program may also be captured. At this stage, however, the metadata is unlikely to be in a form that can be directly exposed to a user – some form of editing will be required before the description of the program can be published.

Content publishing

Once content has been created, the content is then available for publication by a content publisher. This could be, for example, as part of a broadcast service or as a publication on the Internet. The content publishing process defines instantiations of programs – in other words, one output from the content publishing process is information about 'where' the program can be found. In the broadcast case, this means a schedule for the services that are published.

Metadata editing

The metadata editing process takes *raw* information from the content creation and publishing processes and edits this into a form that is suitable for representing the content to the end consumer. The output of this process is edited metadata for the programs and/or metadata describing the location of these programs.

Metadata aggregation

In order to support a given TV-Anytime system, it is likely that metadata from a number of independent content creators and publishers will need to be aggregated. It is important to recognize that the process of metadata aggregation may result in the original metadata being changed.

Metadata publishing

Without prejudice to whether or not a TV-Anytime system is horizontally or vertically integrated, an aggregated metadata set will need to be published to both the content selection and location resolution processes. The content selection process will be largely concerned with the metadata describing programs but may also involve use of the program location metadata. The location resolution service will simply require information about the location of programs.

Content selection

The content selection process may occur through the direct involvement of the consumer or may be performed on the consumer's behalf by a software agent. In order for a software agent to function correctly, metadata describing the consumer and his preferences will need to be provided to the content selection process. This may be either inferred from the consumer 's past history of content selection or by the explicit specification of preferences by the user (or a combination of the two). Note that the content selection process may be, in part, affected by knowledge of the program's location.

Location resolution

The process of location resolution is simply one of discovering where (or when) a program can be found. Details of this discovery process can be found in the TV-Anytime Content Referencing Specification.

The following sections comprise the normative specification of the TV-Anytime metadata system.

4. TV-Anytime Metadata Data Model

Before describing the standard set of TV-Anytime metadata description schemas, we first introduce a simple data modelling methodology that allows us to describe metadata structure at a high level in a manner independent of any particular representation. When describing metadata structure, we make a clear separation between the high-level metadata structure and the specific representation of that structure. The entity-relation diagrammatic syntax used here for describing the model is shown below in Figure 2. This syntax allows relationships between TV-Anytime entities to be clearly stated (*e.g.*, one-to-many), as well as enabling the powerful concept of inheritance, which allows specific types of entity to be derived from generic types.

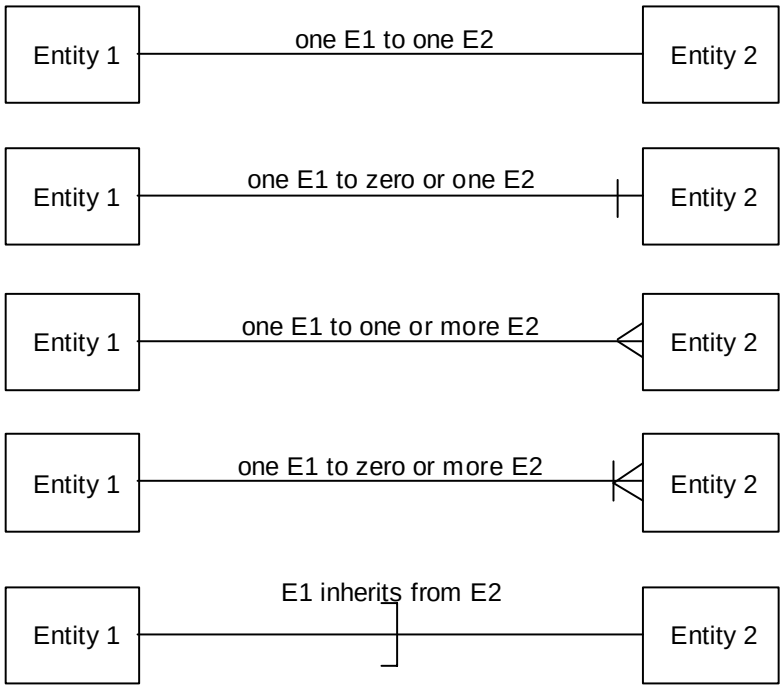

Figure 2 Basic Entity-Relation graph syntax

The cornerstone of TV-Anytime metadata is the CRID, described in TV-Anytime document S-4. As a content reference identifier, the CRID refers to a piece of content, though in some cases it may refer to one or more other CRIDs.

The CRID also acts as the link that connects different content-related metadata descriptions[1] (see Figure 3).

We classify content-related metadata as either *content description metadata* or *instance description metadata.*

As shown in Figure 3, content description metadata is general information about a piece of content that does not change regardless of how the content is published or broadcast. It includes information such as the content's title, textual description, and genre. Typically, the content creator assigns content description metadata before publication.

Instance description metadata describes a particular instance of a piece of content, including information such as the content location, usage rules (pay-per-view, etc.), and delivery parameters (*e.g.*, video format). Instance description metadata is assigned by the content provider as a part of the publication of content. During the search and selection process, a consumer may use both general content and instance descriptions.

[1] Note that different authorities may in fact assign different CRIDs to the same content. Thus, simply comparing CRIDs is not a guaranteed way to determine if two pieces of content are different. Note, however that the TV-Anytime program information DS (Section 5.2.6) allows the attachment of other identifiers in addition to the CRID.

One such identifier that is being standardized through a collaboration of SMPTE, ATSC and ISO is the V-ISAN (Versioned International Audio-visual Number). The goal of the V-ISAN is to provide a central registration authority for all finished audio-visual works. The use of the V-ISAN is not mandatory within the TV-Anytime system, but where present it provides a simple way to determine the relation of two pieces of content. For example, one can determine if two CRIDs refer to the exact same or different versions of the referenced content.

A third category of metadata called consumer metadata includes usage history data (logging data), annotation metadata, and user preferences. Usage history metadata is implicitly generated by the PDR during the course of viewing and browsing content. Bookmarks and notes are examples of annotation metadata which are created explicitly by the consumer.

Figure 3 shows these three types of metadata and how the CRID for an individual content item (*i.e.*, a CRID that does not resolve into further CRIDs) is used to tie them all together. This is not a complete list of all TV-Anytime metadata; only a few representative metadata entities are shown.

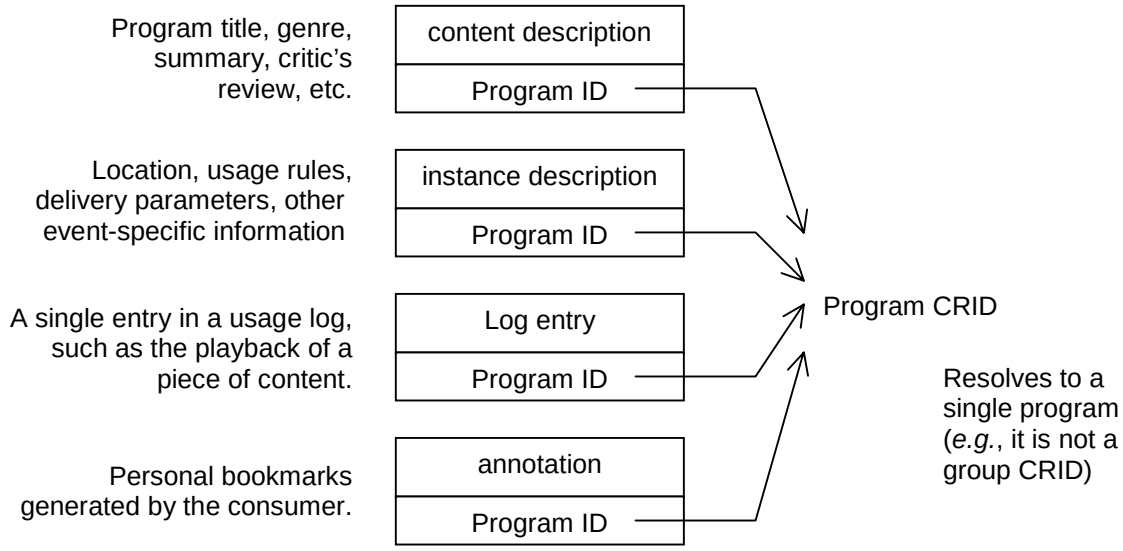

Figure 3 Metadata that references a program CRID

Figure 4 shows some of the major kinds of TV-Anytime metadata and their relationships. Program metadata describes information about single programs, such as the title, genre, etc. A program is defined to be a editorially coherent piece of content which is typically acquired as a whole. The program is referenced via a program CRID (or "leaf CRID"), *i.e.,* a CRID that resolves to a single program.

The same program may be found in any number of locations, as is prescribed by the location resolution process. This relationship is indicated via the one-to-many relationship link from "Program" to "Program Location".

Programs can be grouped into "Program Group" elements such that a group may contain any number of programs, and a program can be a member of any number of groups. Furthermore, program groups themselves can be part of other program groups as depicted in Figure 4. A program group is uniquely identified by a group CRID. Note that as described in the S-4 document, the format of a CRID does not indicate by itself whether that CRID resolves to a program or a list of CRIDs. Several types of program groups are defined in this specification. A third party may define additional program group types.

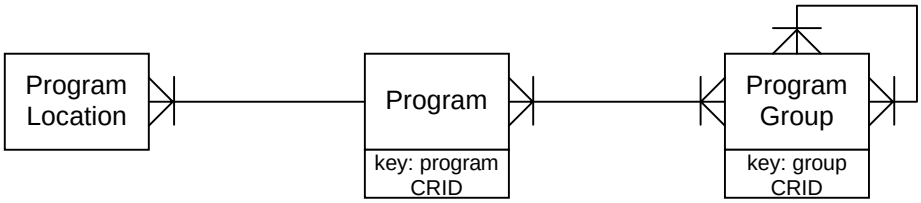

Figure 4 Relationship between major kinds of TV-Anytime metadata

5. Metadata Definitions

For the purpose of interoperability, the TV-Anytime Forum has adopted XML as the common representation format for metadata. XML offers many advantages: it allows for extensibility, supports the separation of data from the application, and is widely used. In addition, powerful XML tools are now available such as XSL (XML Stylesheets), XQL (XML Query Language), and XML databases that can be used to process and manage XML data. As a textual format, XML tends to be rather verbose; however, a number of mechanisms are being developed to reduce the bandwidth when necessary.

A *metadata schema* is the formal definition of the structure and type of metadata. TV-Anytime uses the MPEG-7 Description Definition Language (DDL) [3] to describe metadata structure as well as the XML encoding of metadata. DDL is based on XML schema "W3C Proposed Recommendation " (version 20010306) [1].

5.1 TV-Anytime Namespace

TV-Anytime description schemes that have been developed under the auspices of the TV-Anytime Forum are associated with the TV-Anytime XML namespace. The TV-Anytime namespace is defined as:

```
xmlns="http://www.tv-anytime.org/2001/08/metadata"
```

Note that TV-Anytime metadata also includes description schemes defined by MPEG-7, which use the MPEG-7 namespace as described in [2][2].

The XML specification does not require that an XML instance document include a reference to the schema that describes it; however, for TV-Anytime metadata we do make this requirement in order to identify which schema the document conforms to.

5.2 Content Description Metadata

This section describes metadata that describes content independently of any particular instantiation of that content.

5.2.1 Content Description Requirements

The content description model must be able to represent the following concepts:

1. A simple program

[2] The MPEG-7 data types and description schemes in the TVA Metadata Specification are currently taken from the MPEG-7 MDS Final Committee Draft (FCD) [2]. It is expected that future versions of this specification will refer to the MPEG-7 MDS International Standard. The draft version of the international standard (N4242) will be available at the end of September 2001 at http://cselt.it/mpeg/working_documents.htm.

2. A program with a number of different versions (e.g. edits for sex/violence/language, director's cut, etc.)
3. A program that has been divided into a number of parts for publication (e.g. a 3 hour film shown in 2 parts on different days)
4. A program that is a concatenation of a sequence of other programs (e.g. an omnibus edition of a daily soap opera) – note that this concept is concretely identifiable as a program
5. A series of programs that can be ordered or unordered and bounded or unbounded
6. A collection of series and individual programs that have the same program concept – i.e. a show (e.g. all series of 'Only Fools and Horses' together with the Christmas specials)
7. A publication of a program that may have publication dependent attributes (e.g. a film showing as tribute to a recently deceased actor which would have a different description)

Further to this, the model should attempt to represent these concepts in the simplest form and ideally in a way that reflects the way in which the described content was created in the first place. This should ensure that the information described by the model is naturally understandable both to a person and to agent software acting on behalf of a person.

5.2.2 TV-Anytime Content Description model

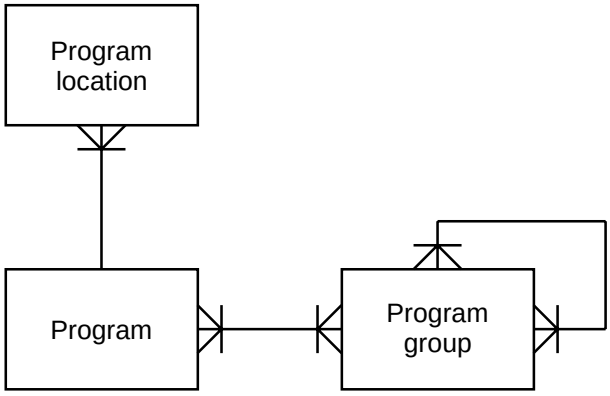

Figure 5 TV-Anytime Content Description Model

Entity definitions

- **Program** - the program represents an editorially coherent piece of content.
- **Program group** - the program group entity simply represents a grouping of programs. A number of different types of group have been identified, such as series, show, aggregate (magazine) program, and program concept. Program groups can also contain other program groups.
- **Program location** A program location contains information about one instance (or "publication event") of a program. Multiple program locations from the same service provider can be grouped to form a schedule.

Relationship definitions

1. **Program to Program location** (zero to many) – a given program can appear at any number of program locations (e.g. schedule events) and a given program location instantiates one program

2. **Program to program group** (many to many) – a given program can be a member of any number of program groups and a given program group can contain any number of programs
3. **Program group to program group** (many to many) – a given arbitrary program group can contain any number of program groups and a given program group can be a member of many program groups.

5.2.3 Basic types

The simple and complex utility types defined below are used throughout the TV Anytime schema specification.

```
<complexType name="IDRefType">
  <attribute name="id" type="Name"/>
</complexType>

<simpleType name="CRIDType">
  <restriction base="anyURI">
    <pattern value="crid://.*/.*"/>
  </restriction>
</simpleType>

<complexType name="CRIDRefType">
  <attribute name="crid" type="tva:CRIDType"/>
</complexType>

<complexType name="FlagType">
  <attribute name="value" type="boolean" use="optional"/>
</complexType>
```

Name	Definition
IDRefType	A complex type that allows a reference to be made to an ID in a description document.
Id	The value of the ID type being referenced
CRIDType	A type to represent a CRID as a URI reference
CRIDRefType	A complex type that allows a reference to be made to a CRID
crid	The value of the CRID being referenced
FlagType	A type that can be used to indicate simple boolean values
value	The value of a boolean flag - can be "true" (default) or "false". The default of true means that a simple <tag/> instance is sufficient to denote a set flag

5.2.4 Description

The following simple and complex types define descriptive attributes of content.

```
<simpleType name="SynopsisLengthType">
  <restriction base="string">
    <enumeration value="short"/>
    <enumeration value="medium"/>
    <enumeration value="long"/>
  </restriction>
</simpleType>

<complexType name="SynopsisType">
  <simpleContent>
    <extension base="string">
      <attribute name="length" type="tva:SynopsisLengthType"
        use="optional"/>
      <attribute ref="xml:lang" use="optional" default="en"/>
    </extension>
  </simpleContent>
</complexType>

<complexType name="RelatedMaterialType">
  <sequence>
    <element name="Format" type="mpeg7:TermUseType" minOccurs="0"/>
    <element name="MediaLocator" type="mpeg7:MediaLocatorType"/>
  </sequence>
</complexType>

<complexType name="CreditsListType">
  <sequence>
    <element name="CreditsItem" type="mpeg7:CreatorType"
      minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="BasicContentDescriptionType">
  <sequence>
    <element name="Title" type="mpeg7:TitleType"
      maxOccurs="unbounded"/>
    <element name="MediaTitle" type="mpeg7:TitleMediaType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="Synopsis" type="tva:SynopsisType" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="Keywords" type="mpeg7:KeywordAnnotationType"
      minOccurs="0" maxOccurs="1"/>
    <element name="Genre" minOccurs="0" maxOccurs="unbounded">
      <complexType>
        <complexContent>
          <extension base="mpeg7:ControlledTermUseType">
            <attribute name="type" default="main">
              <simpleType>
                <restriction base="string">
                  <enumeration value="main"/>
                  <enumeration value="secondary"/>
                  <enumeration value="other"/>
                </restriction>
              </simpleType>
            </attribute>
          </extension>
```

```
      </complexContent>
    </complexType>
  </element>
  <element name="ParentalRating" type="mpeg7:ParentalGuidanceType"
    minOccurs="0" maxOccurs="unbounded"/>
  <element name="Language" type="mpeg7:ExtendedLanguageType"
    minOccurs="0" maxOccurs="unbounded"/>
  <element name="CaptionLanguage" minOccurs="0"
    maxOccurs="unbounded">
    <complexType>
      <simpleContent>
        <extension base="language">
          <attribute name="closed" type="boolean" use="optional"
            default="true"/>
          <attribute name="supplemental" type="boolean"
            use="default" value="false"/>
        </extension>
      </simpleContent>
    </complexType>
  </element>
  <element name="SignLanguage" minOccurs="0" maxOccurs="unbounded">
    <complexType>
      <simpleContent>
        <extension base="language">
          <attribute name="primary" type="boolean" use="optional"/>
          <attribute name="translation" type="boolean"
            use="optional"/>
          <attribute name="type" type="string" use="optional"/>
        </extension>
      </simpleContent>
    </complexType>
  </element>
  <element name="CreditsList" type="tva:CreditsListType"
    minOccurs="0"/>
  <element name="RelatedMaterial" type="tva:RelatedMaterialType"
    minOccurs="0" maxOccurs="unbounded"/>
  <element name="ProductionYear" type="positiveInteger"
    minOccurs="0" maxOccurs="unbounded"/>
  <element name="ProductionCountry" type="mpeg7:countryCode"
    minOccurs="0" maxOccurs="unbounded"/>
  <element name="CreationLocation" type="mpeg7:PlaceType"
    minOccurs="0" maxOccurs="unbounded"/>
  <element name="CreationDate" type="date" minOccurs="0"/>
  <element name="ReleaseInformation" minOccurs="0"
    maxOccurs="unbounded">
    <complexType>
      <sequence>
        <element name="ReleaseDate" type="date"/>
        <element name="ReleaseCountry" type="mpeg7:countryCode"/>
      </sequence>
    </complexType>
  </element>
 </sequence>
</complexType>
```

Name	Definition
SynopsisLengthType	An enumeration of the possible values of the length qualifier for a synopsis. The possible values of this

	enumerated type are as follows: • *short* – the length of the synopsis will not exceed 90 characters • *medium* – the length of the synopsis will not exceed 180 characters • *long* – the length of the synopsis will not exceed 1200 characters
SynopsisType	A complex type to define a synopsis
length	The length of the synopsis. This attribute is optional. If no length is specified, then the synopsis may be of any length.
RelatedMaterialType	A complex type that refers to other media assets that are related to the AV content (e.g. program) that is described
Format	Specifies the type (e.g. file format) of the media asset (optional). The format can either be specified as a free term, or chosen from the MPEG-7 "MPEG7FileFormatCS" classification scheme listed in section 17.1.2.1.1 of [2].
MediaLocator	Specifies the location of the media asset. Defined as an MPEG-7 datatype, MediaLocatorType (See Sec. 6.4.1 of [2] for a detailed description).
CreditsListType	A complex type that defines a list of credits for the specified program
CreditsItem	Describes a member in the list of credits. Defined as an MPEG-7 datatype, CreatorType (See Sec. 9.1.2.1 2 of [2] for a detailed description). Note that this element can be a reference to a person in the CreditsInformationTable.
BasicContentDescriptionType	A complex type that defines standard program description elements.
Title	A title of the program. A program can have multiple titles, e.g. in different languages. Defined as an MPEG-7 datatype, TitleType (See Sec. 9.1.2 in [2] for a detailed specification).
MediaTitle	A media asset (e.g. image) that can be used as a 'title' for a program. Content that is not part of the original program can be specified and used as a (promotional) AV title. Defined as an MPEG-7 datatype, TitleMediaType (See Sec. 9.1.2 in [2] for a detailed specification).
Synopsis	A textual description of the program.
Keywords	A list of keywords for the program. A keyword can be a single word or an entire phrase made up of multiple words. Defined as an MPEG-7 datatype, KeywordAnnotationType (See Sec. 7.2.4 of [2] for a detailed specification).
Genre	A genre for the program. The thesaurus in Appendix B defines the normative TV-Anytime set of genres.

`type`	Indicates the type of the genre of the multimedia content. The types of genres are defined as follows. *main* – The specified genre is the main, or primary. This is the default value. *secondary* – The specified genre is a secondary genre, such as a subgenre. *other* – The specified genre is an alternative genre, such as one defined or used by 3rd parties.
`ParentalRating`	A parental rating code for the program. Defined as an MPEG-7 datatype, `ParentalGuidanceType` (See Sec. 9.1.3 of [2] for a detailed specification).
`Language`	Describes one spoken language for the program. There may be more than one spoken language specified for a program.
`CaptionLanguage`	Describes one language of the caption information included with the program. The type of the caption information associated with the program is denoted by the closed attribute. Closed captions can be turned on or off by the user, while open captions (or subtitles) are part of the picture itself and remain visible.
`closed`	Indicates whether the specified caption is closed. Default value of the attribute is true; if the attribute is set to false, then the provided caption description refers to open captions/subtitles.
`supplemental`	Indicates whether the captions provide descriptions of the scene for the benefit of hearing or visually impaired, in addition to a direct translation of the spoken words. Closed captions may include such descriptive information, such as speaker identification, and non-speech sounds that would be missed.
`SignLanguage`	Specifies the audio sign language provided for the multimedia content, and, optionally, qualifies the use of signing as a primary language and/or as a translation of the spoken dialogue.
`primary`	Indicates if the sign language is the primary language of the content or not, i.e, if the content is produced specifically for the hearing impaired or not.
`translation`	Indicates if the sign language is a translation of the spoken dialogue or not.
`type`	Indicates the type (i.e. country of origin) of the specified sign language
`CreditsList`	The list of credits (e.g. actors, directors, etc.) for the program
`RelatedMaterial`	A reference to any other material related to a program
`ProductionYear`	The date (year) when the program was produced

`ProductionCountry`	The country in which the program was produced. Defined as an MPEG-7 datatype, `countryCode`.
`CreationLocation`	The location where the program was created. Defined as an MPEG-7 datatype, `PlaceType` (See Sec. 7.5.1 of [2] for a detailed specification).
`CreationDate`	The date the program was created
`ReleaseInformation`	Information about the country and date of release of a program. Some metadata providers may want to publish this information as part of the content description.
`ReleaseDate`	The date when the program was released.
`ReleaseCountry`	The country where the program was released. Defined as an MPEG-7 datatype, `countryCode`.

5.2.5 Audio and video information

The following simple and complex types define technical attributes of audio and video.

```
<complexType name="AVAttributesType">
  <sequence>
    <element name="FileFormat" type="mpeg7:ControlledTermUseType"
      minOccurs="0"/>
    <element name="FileSize" type="nonNegativeInteger"
      minOccurs="0"/>
    <element name="System" type="mpeg7:ControlledTermUseType"
      minOccurs="0"/>

    <element name="BitRate" minOccurs="0">
      <complexType>
        <simpleContent>
          <extension base="nonNegativeInteger">
            <attribute name="variable" type="boolean" default="false"/>
            <attribute name="minimum" type="nonNegativeInteger"
              use="optional"/>
            <attribute name="average" type="nonNegativeInteger"
              use="optional"/>
            <attribute name="maximum" type="nonNegativeInteger"
              use="optional"/>
          </extension>
        </simpleContent>
      </complexType>
    </element>

    <element name="AudioAttributes" minOccurs="0">
      <complexType>
        <sequence>
          <element name="Coding" type="mpeg7:ControlledTermUseType"
            minOccurs="0"/>
          <element name="NumOfChannels" type="positiveInteger"
            minOccurs="0"/>
          <element name="MixType" type="mpeg7:ControlledTermUseType"
            minOccurs="0"/>
        </sequence>
      </complexType>
    </element>
```

```
   <element name="VideoAttributes" minOccurs="0">
     <complexType>
        <sequence>
        <element name="Coding" type="mpeg7:ControlledTermUseType"
          minOccurs="0"/>
        <element name="Scan" type="tva:ScanType" minOccurs=”0”/>
        <element name="HorizontalSize" type="positiveInteger"
          minOccurs=”0"/>
        <element name="VerticalSize" type="positiveInteger"
          minOccurs=”0”/>
        <element name="AspectRatio" type="tva:AspectRatioType"
          minOccurs=”0”/>
        <element name="Color" type="tva:ColorType" minOccurs="0"/>
       </sequence>
   </complexType>
   </element>
  </sequence>
</complexType>

<simpleType name="ScanType">
  <restriction base="string">
    <enumeration value="interlaced"/>
    <enumeration value="progressive"/>
  </restriction>
</simpleType>

<simpleType name="ColorTypeType">
  <restriction base="string">
    <enumeration value="color"/>
    <enumeration value="blackAndWhite"/>
    <enumeration value="blackAndWhiteAndColor"/>
    <enumeration value="colorized"/>
  </restriction>
</simpleType>

<complexType name="ColorType">
  <attribute name="type" type="tva:ColorTypeType"/>
</complexType>

<simpleType name="AspectRatioType">
  <restriction base="string">
    <pattern value="\d+:\d+"/>
  </restriction>
</simpleType>
```

Name	**Definition**
`AVAttributesType`	A complex type defining a set of elements that can describe technical attributes of either audio or video or both.
`FileFormat`	Describes the file format of the program instance.
`FileSize`	Indicates the size, in bytes, of the file where the program instance is stored.
`System`	Describes the broad media format of the program instance. This term should be taken from the MPEG-7 “MPEG7SystemCS” classification scheme

	listed in section 17.1.2.3.1 of [2].
BitRate	Indicates the nominal bit rate in bits/s of the program instance.
variable	Indicates whether the BitRate is variable or fixed. If the BitRate is variable, three optional attributes can be used to specify the minimum, maximum and average bitrates.
minimum	Indicates the minimum numerical value for the BitRate in case of variable bit rate.
average	Indicates the average numerical value for the BitRate in case of variable bit rate.
maximum	Indicates the maximum numerical value for the BitRate in case of variable bit rate.
AudioAttributes	A complex type defining a set of elements that describe audio characteristics
Coding	The coding format of the audio. This term should be taken from the MPEG-7 "MPEG7AudioCodingFormatCS " classification scheme listed in section 17.1.2.6.1 of [2].
NumOfChannels	The number of channels of audio
MixType	The type of the audio mix. This term should be taken from the MPEG-7 "MPEG7AudioPresentationCS" ClassificationScheme listed in section 17.1.2.5.1 of [2].
VideoAttributes	A complex type defining a set of elements that describe video characteristics
Coding	The coding format of the video. This term should be taken from the MPEG-7 "MPEG7VisualCodingFormatCS " classification scheme listed in section 17.1.2.4.1 of [2].
Scan	The scan type of the video
HorizontalSize	The horizontal size in pixels of the video
VerticalSize	The vertical size in pixels of the video
AspectRatio	The aspect ratio of the video
Color	The color format of the video (e.g. black and white)
ScanType	A simple enumerated type defining the allowable values of the ScanType element above. ScanType can take on the value interlaced or progressive.
ColorTypeType	A simple enumerated type defining the allowable values of the Color element above. Allowed values are: • color – the content was produced using a color video format • blackAndWhite – the content was produced using a black and white video format • blackAndWhiteAndColor – the content

	contains a mixture of video that was originally produced in color and content that was produced in black and white • `colorized` – the content was originally produced using a black and white video format, and color was added after original production
`ColorType`	A complex type that describes color format
`type`	The type of color format
`AspectRatioType`	A simple string type which allows aspect ratios to be specified in the form 'h:v' where h and v represent horizontal and vertical dimensions respectively

5.2.6 Program information

```
<element name="ProgramInformation"
type="tva:ProgramInformationType"/>

<complexType name="ProgramInformationType">
  <sequence>
    <element name="BasicDescription"
      type="tva:BasicContentDescriptionType"/>
    <element name="OtherIdentifier" type="mpeg7:UniqueIDType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="AVAttributes" type="tva:AVAttributesType"
      minOccurs="0"/>
    <element ref="tva:MemberOf" minOccurs="0" maxOccurs="unbounded"/>
    <element name="OmnibusOf" minOccurs="0">
      <complexType>
        <sequence>
          <element name="AggregatedProgram" type="tva:CRIDRefType"
            minOccurs="2" maxOccurs="unbounded"/>
        </sequence>
      </complexType>
    </element>
  </sequence>
  <attribute name="programId" type="tva:CRIDType"/>
  <attribute name="version" type="integer"/>
</complexType>
```

Name	Definition
`ProgramInformation`	An element that instantiates the ProgramInformationType
`ProgramInformationType`	A complex type that describes a program
`BasicDescription`	The description of the program
`OtherIdentifier`	A code that can be used in addition to the CRID to identify a piece of content (e.g. a V-ISAN)
`AVAttributes`	Audio-visual attributes that are applicable to the program as originated
`MemberOf`	A list of groups of which the program is a member

OmnibusOf	An optional list of programs for which this program is an 'omnibus edition'
AggregatedProgram	A program that has been aggregated into an 'omnibus edition'
programId	The CRID for the program
version	A version number for the information describing this program

5.2.7 Group Information

```
<simpleType name="GroupTypeType">
  <restriction base="string">
    <enumeration value="series"/>
    <enumeration value="show"/>
    <enumeration value="programConcept"/>
    <enumeration value="magazine"/>
    <enumeration value=”programCompilation”/>
    <enumeration value="otherCollection"/>
    <enumeration value="otherChoice"/>
  </restriction>
</simpleType>

<element name="GroupInformation" type="tva:GroupInformationType"/>

<complexType name="GroupInformationType">
  <sequence>
    <element name="BasicDescription"
      type="tva:BasicContentDescriptionType"/>
    <element name="MemberOf" type="tva:GroupRefType" minOccurs="0"
      maxOccurs="unbounded"/>
  </sequence>
  <attribute name="groupId" type="tva:CRIDType"/>
  <attribute name="groupType" type="tva:GroupTypeType"
    use="optional"/>
  <attribute name="ordered" type="boolean" use="optional"/>
  <attribute name="numOfItems" type="positiveInteger"
    use="optional"/>
  <attribute name="version" type="integer"/>
</complexType>

<element name="MemberOf" type="tva:GroupRefType"/>
<complexType name="GroupRefType">
  <complexContent>
    <extension base="tva:CRIDRefType">
      <attribute name="index" type="positiveInteger" use="optional"/>
    </extension>
  </complexContent>
</complexType>

<element name="EpisodeOf" type="tva:GroupRefType"
substitutionGroup="tva:MemberOf"/>

<simpleType name="DerivationReasonType">
  <restriction base="string">
    <enumeration value="violence"/>
    <enumeration value="language"/>
```

```
    <enumeration value="sex"/>
    <enumeration value="duration"/>
    <enumeration value="other"/>
  </restriction>
</simpleType>

<element name="DerivedFrom" substitutionGroup="tva:MemberOf">
  <complexType>
    <complexContent>
      <extension base="tva:GroupRefType">
        <attribute name="derivationReason"
          type="tva:DerivationReasonType" use="optional"/>
      </extension>
    </complexContent>
  </complexType>
</element>
```

Name	Definition
GroupTypeType	A simple enumerated type defining the permitted types of group. The allowed values for this field are as follows: *series* – an ordered or unordered collection of programs that is shown in a sequence (e.g. "Friendz" season 1). An unbounded series (e.g. an ongoing drama series) may be considered to be a serial *show* – a program theme that is typically be associated with a collection of series (e.g. all episodes of Friends) *programConcept* – the editorial concept for a program from which specific program versions have been derived (e.g. the concept of "Blood Runner" as opposed to "Blood Runner – The Director's Cut" as a specific version of that concept) *magazine* – a collection of individual programs that are shown as a group because they are editorially coherent (e.g. a general sports program with individual sub-programs covering different events) *programCompilation* – a collection of programs that is used to allow segments from multiple programs to be combined in segment groups. When used in conjunction with segmentation information, a programCompilation program group allows, for example, several related news segments from different news programs to be grouped for playback in sequence. *otherCollection* – can be used for any group not defined in the preceding list where all members of the group should be acquired if the group is selected *otherChoice* – can be used for any group not defined in the list above where only one member

Name	Definition
	of the group should be acquired is the group is selected
GroupInformation	An element that instantiates a GroupInformationType
GroupInformationType	A complex type to describe a group
BasicDescription	The description of the group
MemberOf	A list of other groups of which this group is a member. Note: MemberOf is a substitution group that includes EpisodeOf and DerivedFrom
groupId	A unique CRID that identifies the group
groupType	The type of the group (e.g. series)
ordered	Optional boolean flag that indicates whether or not the group is ordered
numOfItems	Optional indication of the total number of members in the group. This is of significance for series where an episode needs to referred to as episode # of n
version	The version of the group. Larger version numbers indicate later versions.
MemberOf	Element that instantiates the GroupRefType. This is used as a substitution group for a variety of different types of group reference
GroupRefType	A complex type, based on CRIDReferenceType, that references a group
index	An index for the program within the specified group. This would be used, for example, to specify an episode number for a program in a series
EpisodeOf	An equivalent element to MemberOf that indicates membership of a series
DerivationReasonType	An enumerated list of permitted reasons for deriving a program version from a program concept. Permitted values are violence, language, sex, duration and other.
DerivedFrom	An equivalent element to MemberOf that indicates that a program version has been derived from a generic program concept. Note: this element cannot be used in the context of a GroupInformationType
derivationReason	The reason for the derivation of the program version.

5.2.8 Media Review DS

The MediaReview DS includes third party reviews of an AV content, for example, critic's reviews of a movie. The MediaReview DS may be used in conjunction with user preferences to enable the automatic selection, filtering or recording of AV content on a TVA device.

The `MediaReview` DS in the TVA set of metadata schemas is based on the `MediaReview` description scheme as defined in section 9.1.3 of [2].

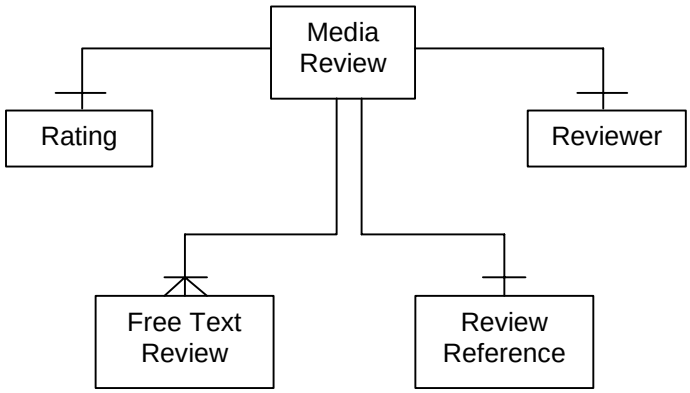

Figure 6 Entity-Relationship Graph for the Media Review DS

Additionally, TV-Anytime defines the following schemas for dealing with `MediaReview` instances.

```
<element name="ProgramReviewTable"
type="tva:ProgramReviewTableType"/>

<complexType name="ProgramReviewTableType">
  <sequence>
    <element ref="tva:ProgramReviews" minOccurs="0"
    maxOccurs="unbounded"/>
  </sequence>
  <attribute name="version" type="integer"/>
</complexType>
```

Name	*Definition*
`ProgramReviewTable`	An element that instantiates the `ProgramReviewTableType`
`ProgramReviewTableType`	A complex type that tabulated descriptions of reviews associated with (multiple) programs
`ProgramReviews`	Describes the reviews associated with a single program
`version`	A version number for the information in this table

```
<element name="ProgramReviews" type="tva:ProgramReviewsType"/>

<complexType name="ProgramReviewsType">
  <sequence>
    <element name="Program" type="tva:CRIDRefType"/>
    <element name="Review" type="mpeg7:MediaReviewType"
    maxOccurs="unbounded"/>
  </sequence>
  <attribute name="version" type="integer"/>
</complexType>
```

Name	*Definition*
ProgramReviews	An element that instantiates the ProgramReviewsType
ProgramReviewsType	A complex type that describes (multiple) review(s) of a program
Program	A reference to the program using a CRID
Review	A review of the program; there can be multiple reviews
version	A version number for the review information describing the program

5.2.9 Basic Metadata (Informative)

We have defined the descriptive metadata that can be associated with content above. Because TV-Anytime metadata will be processed on a variety of devices, including devices with extremely limited resources, we classify the above metadata into required, recommended and optional metadata elements.

5.2.9.1 Mandatory

Name	**Requirement**
Title	All Program Information and Group Information objects shall contain a meaningful Title field.

5.2.9.2 Recommended

This is a preliminary list of recommended metadata elements to be reviewed at a later date in with respect to application-specific guidelines.

Name	**Guideline**
Synopsis	It is recommended that all Program Information and Group Information objects contain a meaningful Synopsis element.
Genre	It is recommended that all Program Information and Group Information objects contain a meaningful Genre element.
Languages	It is recommended that all Program Information and Group Information objects contain a meaningful Languages element to define the spoken, subtitle and audio description properties of the content.
MemberOf	It is recommended that Program Information and Group Information objects shall use the MemberOf element. Links to Group Information are one the key methods enabled by TV Anytime for content discovery.

5.2.10 Optional Metadata (Informative)

All other metadata defined in this section are optional.

5.3 Instance Description Metadata

In the previous section, we dealt with *content description* metadata, which associates metadata with a piece of content. The key for linking content metadata to content is the CRID. In this section, we describe *instance description* metadata. Instance description metadata is useful in cases where there are meaningful differences between instances of the same content (that is, instances of content that share the same CRID). Instance description metadata is linked to a particular event-related instance of content..

5.3.1 Program location entities

A program location contains information about one instance (or "publication event") of a program. Multiple program locations from the same service provider can be grouped to form a schedule.

A metadata provider aggregates a set of program locations (e.g. schedules) into a `ProgramLocationTable`, as described in Section 5.6.1, and includes this table in a TV-Anytime metadata instance document.

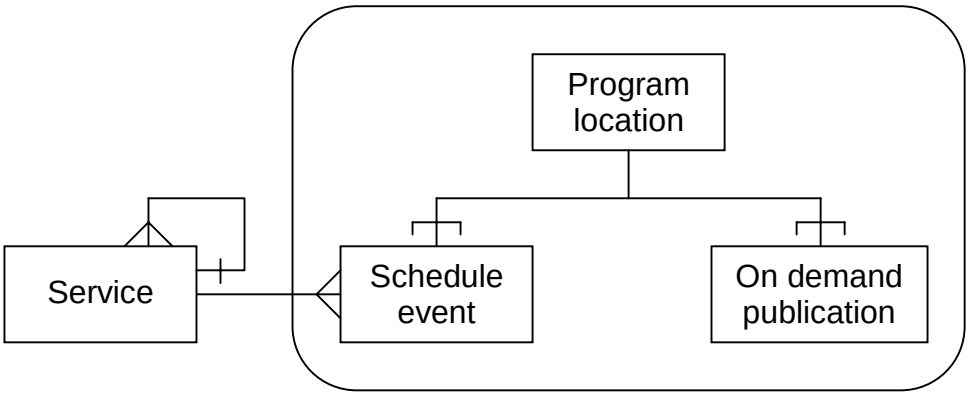

Figure 7 Single Program Location Data Model

Figure 7 depicts the high-level data model for program location.

Entity definitions

- **Program location** - the program location represents a generic program location, regardless of the nature of the medium it addresses - two obvious examples being broadcast services and the Web. The principle feature of a program location is that it may 'contain' at most one program.
- **Schedule event** - the schedule event is a specific type of program location that is appropriate for describing broadcast program locations. The schedule event associates a given broadcast location (service, time and duration) with a given program.
- **Service** - the service entity represents a distinct (according to content) stream of broadcast material. A service is carried in some form of physical channel but the two entities are not synonymous as a given service can be broadcast on a variety of physical channels.

For syntactic convenience, TV-Anytime provides a mechanism to group a series of schedule events from the same provider using the `ScheduleType`. The `ScheduleType` allows the metadata provider to specify a given service just once, and then provide a list of schedule events associated with that service.

5.3.2 Program Location

Figure 8 is useful in understanding how the abstract model above has been implemented in this specification.

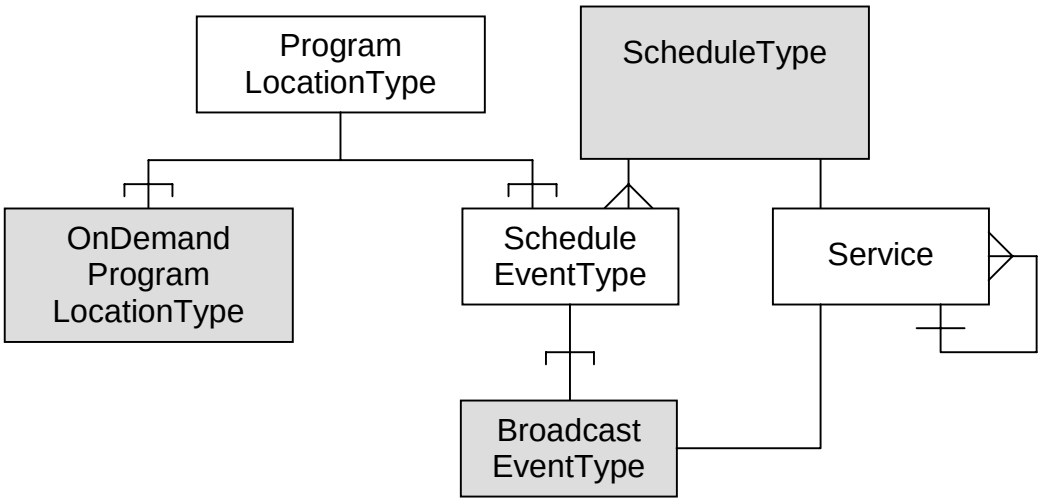

Figure 8 `ProgramLocationType` and related types

Figure 8 shows the `ProgramLocationType`, and a number of related types. The types in shaded boxes may be used as entries in a program location table (see section 5.6.1).

`ProgramLocationType` is an abstract type that represents one EPG entry. Derived types are `OnDemandProgramLocationType` and `BroadcastEventType`. The definition and semantics of each type are described below.

```
<complexType name="EventInformationType">
  <sequence>
    <element name="PublishedTime" type="dateTime" minOccurs="0"/>
    <element name="PublishedDuration" type="duration" minOccurs="0"/>
    <element name="Live" type="tva:FlagType" minOccurs="0"/>
    <element name="Repeat" type="tva:FlagType" minOccurs="0"/>
    <element name="FirstShowing" type="tva:FlagType" minOccurs="0"/>
    <element name="LastShowing" type="tva:FlagType" minOccurs="0"/>
    <element name="Free" type="tva:FlagType" minOccurs="0"/>
    <element name="PPV" type="tva:FlagType" minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="ProgramLocationType" abstract="true ">
  <complexContent>
    <sequence>
      <element name="Program" type="tva:CRIDRefType"/>
      <element name="InstanceDescription"
        type="tva:InstanceDescriptionType" minOccurs="0"/>
    </sequence>
  </complexContent>
</complexType>

<complexType name="ScheduleType">
  <complexContent>
    <sequence>
      <element name="Event" type="tva:ScheduleEventType" minOccurs="1"
        maxOccurs=”unbounded”/>
      <element name="ServiceId" type="tva:IDRefType"/>
    </sequence>
  </complexContent>
```

```
</complexType>

<complexType name="ScheduleEventType">
  <complexContent>
    <extension base="tva:ProgramLocationType">
      <sequence>
        <element name="BroadcastURL" type="anyURI" minOccurs="0"/>
        <element name="EventDescription"
          type="tva:EventInformationType" minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="BroadcastEventType">
  <complexContent>
    <extension base="tva:ScheduleEventType">
      <sequence>
        <element name="ServiceId" type="tva:IDRefType"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="OnDemandProgramPublicationType">
  <complexContent>
    <extension base="tva:ProgramLocationType">
      <sequence>
        <element name="StartOfAvailability" type="dateTime"
          minOccurs="0"/>
        <element name="EndOfAvailability" type="dateTime"
          minOccurs="0"/>
        <element name="URL" type="anyURI"/>
        <element name="FirstAvailability" type="tva:FlagType"
          minOccurs="0"/>
        <element name="LastAvailability" type="tva:FlagType"
          minOccurs="0"/>
      </sequence>
    </extension>
  </complexContent>
</complexType>

<complexType name="InstanceDescriptionType">
  <sequence>
    <element name="Title" type="mpeg7:TitleType" minOccurs="0"/>
    <element name="Synopsis" type="tva:SynopsisType" minOccurs="0"/>
    <element name="AVAttributes" type="tva:AVAttributesType"
      minOccurs="0"/>
  </sequence>
  <attribute name="version" type="integer"/>
</complexType>
```

Name	Definition
`EventInformationType`	A complex type that provides descriptive information about a broadcast event
`PublishedTime`	The time at which the program is advertised as starting. Note that this will typically be different

	from the actual exact start time. The precise start time is provided by the location resolution mechanism, in the form of a locator.
PublishedDuration	The advertised duration of the program. The actual duration is provided by the location resolution mechanism, in the form of a locator.
Live	A flag to indicate if the program is a live broadcast
Repeat	A flag to indicate if the program is a repeat
FirstShowing	A flag to indicate if this instance is a 'first showing'
LastShowing	A flag to indicate if this instance is a 'last showing'. Typically this will be used for film services that repeat films over a given period
Free	A flag to indicate if access to this instance of the program is free
PPV	A flag to indicate if this instance of the program is a 'Pay Per View' event
ProgramLocationType	An abstract type that represents a single program.
Program	A reference to the CRID that this description describes.
InstanceDescription	Descriptive metadata about this instance of content. Instance metadata is mostly comprised of technical information such as encoding formats; however, a particular instance may also include a synopsis that overrides any synopsis that might have been defined in a corresponding ProgramInformation instance.
BroadcastEvent	An element that instantiates BroadcastEventType and is equivalent to a ProgramLocation
BroadcastEventType	A complex type derived from ScheduleEventType that allows individual events to be described outside the context of a schedule (i.e. where the service cannot be inferred)
ServiceId	A reference to the service carrying the event
Schedule	An element that instantiates MultipleProgramLocationType and is equivalent to a ProgramLocation.
ScheduleType	A complex type derived representing a series of schedule events that are associated with one service.
Event	A list of schedule events
ServiceId	The id of the service that carries the schedule events
ScheduleEventType	A complex type derived from ProgramLocationType that describes a broadcast event that is part of a schedule (*i.e.*,

	where the service is already known). Note that instances of `ScheduleEventType` will always be included in a `Schedule` instance.
`BroadcastURL`	An optional broadcast URL for the event (*e.g.,* a DVB URL). This URL is non-authoritative. Authoritative location information always comes from the location resolution system.
`EventDescription`	The description of the event (that is, a description of the publication of the program, not the program itself)
`OnDemandProgramPublicationType`	A complex type derived from `ProgramLocationType` used to describe instances that can be acquired on demand (as opposed to broadcast).
`StartOfAvailability`	The time and date that this program will first be available.
`EndOfAvailability`	The time and date that this program will no longer be available.
`URL`	The URL that defines the location of the program
`FirstAvailability`	True if this publication is the first publication of the content, false otherwise.
`LastAvailability`	True if this publication is the last publication of the content, false otherwise.
`Program`	A mandatory reference to the program being instantiated
`InstanceDescriptionType`	Complex type used to describe program instances.
`Title`	A title of the program. An instance of a program can have a different title. Defined as an MPEG-7 datatype, `TitleType` (See Sec. 9.1.2 in [2] for a detailed specification). When this element exists, it completely overrides any `Title` that might exist for the corresponding `ProgramInformation` object.
`Synopsis`	A textual description of this instance. Typically, the synopsis for a program will be described in the `ProgramInformation` type, and the instance description will not contain a synopsis. However, in some cases the metadata provider may wish to supply a synopsis for a particular instance of content that includes event-specific information (for example, a showing of a film that is a tribute to a recently deceased director). When this element exists, it completely overrides any `Synopsis` that might exist for the corresponding `ProgramInformation` object.
`AVAttributes`	Technical attributes about this particular instance.
`Version`	A version number for the instance description

NOTE: TV-Anytime is currently considering the possibility of introducing a locator identifier (replacing the currently removed InstanceID) of URI type as an element in the ProgramLocationType. The purpose of this modification would be to complement the current CRID resolving mechanism in order to support the capture of content from a specific location. The locator identifier should also be used as the link between a location and its associated metadata.

Draft Semantic for the LocatorID, if required: A locator identifier shall identify a particular location related to a CRID. The locator identifier shall be unique within the CRID domain and have the same life cycle as the CRID.

5.3.3 Service information

```
<complexType name="ServiceRefType">
  <sequence>
    <element name="ValidPeriod" minOccurs="0" maxOccurs=”unbounded”>
      <complexType>
        <sequence>
          <element name="ValidFrom" type="timeDate" minOccurs="0"/>
          <element name="ValidTo" type="timeDate" minOccurs="0"/>
        </sequence>
      </complexType>
    </element>
    <element name="ServiceId" type="tva:IDRefType"/>
  </sequence>
</complexType>

<element name="ServiceInformation"
type="tva:ServiceInformationType"/>

<complexType name="ServiceInformationType">
  <sequence>
    <element name="ParentService" type="tva:ServiceRefType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="Name" type="string"/>
    <element name="Owner" type="string"/>
    <element name="Logo" type="mpeg7:MediaLocatorType" minOccurs="0"
      maxOccurs="unbounded"/>
  </sequence>
  <attribute name="serviceId" type="ID"/>
</complexType>
```

Name	Definition
`ServiceRefType`	A complex type that allows a reference to be made to a service
`ValidPeriod`	An optional time window that can be applied to the reference. If only `ValidFrom` is specified, then the service reference is assumed to be valid any time after `ValidFrom`. If only `ValidTo` is specified, then the service reference is assumed to be valid any time up until the `ValidTo` time. (In some regions, the same physical channel is allocated to more than one service. Thus, multiple service “timeshare” the same channel. In such cases, `ValidPeriod` can be used to describe the

	time period during which a service is valid.)
ValidFrom	Start time and date from which the reference is valid
ValidTo	End time and date from which the reference is valid
ServiceId	The service that is being referenced
ServiceInformation	An element that instantiates ServiceInformationType
ServiceInformationType	A complex type that allows a service to be described
ParentService	A reference to a parent service when the service being described inherits a part of its schedule from another service (e.g. regional variations from a national service). Note that multiple parent services may be specified on a time exclusive basis
Name	The name of the service
Owner	The brand owner of the service
Logo	A network logo, such as an image or jingle.
ServiceId	The unique ID for the service

5.4 Consumer Metadata

5.4.1 Usage History DS

This section presents a description scheme for describing usage history information gathered over extended periods of time. The collected usage history provides a list of the actions carried out by the user for an observation period, which can subsequently be used by automatic analysis methods to generate user preferences.

A standardized format for exchange of usage history information is important for ensuring interoperability between various devices and platforms. Collection and representation of usage history information in a standardized format are relevant to various application areas and usage scenarios identified by the TV-Anytime Forum, which include the following:

- Tracking and monitoring the content viewed by individual members of a household.
- Building a personalized TV guide by tracking user viewing habits.
- Selling viewing history to advertisers.
- Tracking and monitoring content usage for more efficient content development.
- Selling of usage data by service provider.
- Compensating the user for making his/her usage history data available to content providers.

The TV-Anytime Forum Summary schema is based on the UsageHistory DS as specified in [2], section 15.2.

Figure 9 depicts the entity-relation diagram for the UsageHistory description scheme.

A description instance contains a UserIdentifier element, which specifies the user or the group of users whose content consumption information is provided. The usage history is specified by the UserActionHistory Description Scheme, which contains multiple lists of the actions performed by the user over an observation period. Note that multiple, non-overlapping observation periods can be specified for an action list. Each action list is action-type specific; *i.e.*, a single list contains actions of a certain type (such as "play" or "record") only. The specific types of actions that are tracked (i.e. the values allowed for the ActionType element)

are defined as members of a dictionary/thesaurus[3], which enables new types of actions to be supported in the future (by augmenting the dictionary). Associated with every user action are the time of the action, the CRID of the program for which the action took place, and optional referencing elements which allow related links or resources about the action to be provided. It is assumed that descriptions for the programs cited in the action history are readily accessible through the provided content reference IDs.

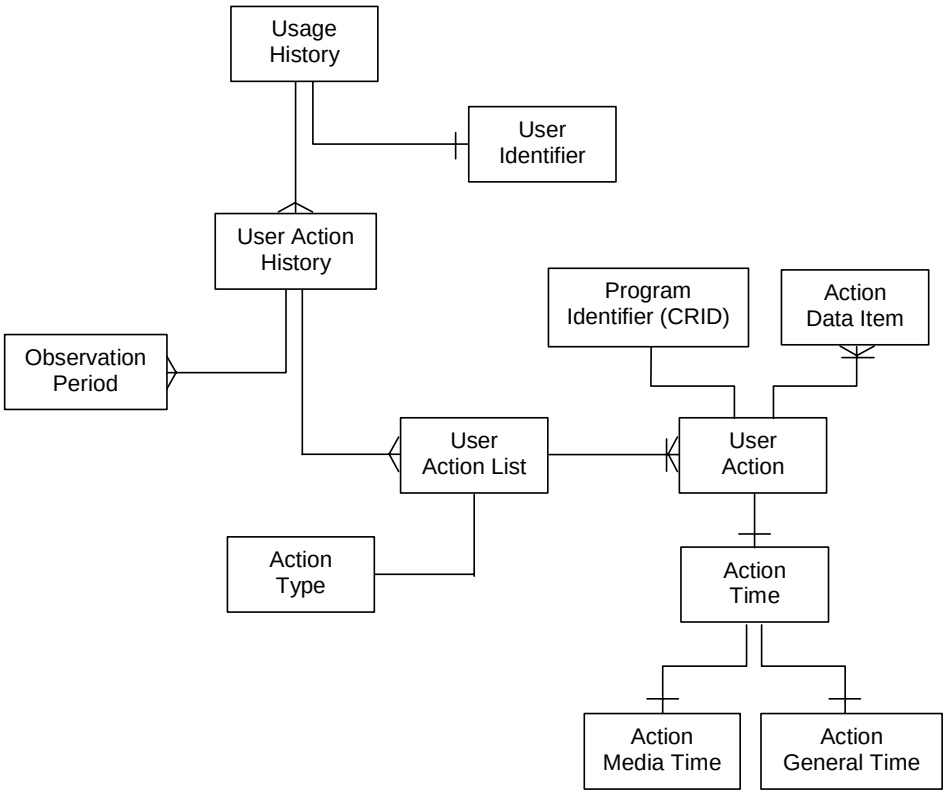

Figure 9 Usage History description scheme

The following sections contain a specification of the syntax and semantics of the UsageHistory description scheme.

5.4.1.1 Usage History DS

The UsageHistory DS describes the audiovisual content consumption history for a user, as lists of the actions performed by the user over an observation period.

The specification of the UsageHistory DS is given in section 15.2.1 of [2]. Syntax of the UsageHistory DS is specified in section 15.2.1.1 in [2]. Semantics of the Usage History DS are specified in section 15.2.1.2 in [2].

5.4.1.2 UserActionHistory DS

The UserActionHistory DS describes multiple user action lists, each of which provides a temporally ordered log of a specific type of user action, such as “Record” or “Play,” regarding audiovisual content.

[3] This thesaurus is provided in Appendix A.

The specification of the UserActionHistory DS is given in section 15.2.2 of [2]. Syntax of the UserActionHistory DS is specified in section 15.2.2.1 in [2]. Semantics of the UserActionHistory DS are specified in section 15.2.2.2 in [2].

5.4.1.3 UserActionList DS

The UserActionList description scheme specifies a structured list of user action items, organized according to action type. Every UserAction is associated with a single program or content entity only. A TV-Anytime dictionary of valid user actions is provided in Appendix A.

The specification of the UserActionList DS is given in section 15.2.3 of [2]. Syntax of the UserActionList DS is specified in section 15.2.3.1 in [2]. Semantics of the UserActionList DS are specified in section 15.2.3.2 in [2].

5.4.1.4 UserAction DS

The UserAction description scheme provides detailed information about individual user actions, including the time of occurrence, duration, associated CRID, and references to related content descriptions and material.

The specification of the UserAction DS is given in section 15.2.4 of [2]. Syntax of the UserAction DS is specified in section 15.2.4.1 in [2]. Semantics of the UserAction DS are specified in section 15.2.4.2 in [2].

In order to ensure full compliance with the CRID definition as stated by the TVA Content Referencing Specification, the `ProgramIdentifier` element of the `UserAction` description scheme is constrained as follows:

- If the `type` attribute of `ProgramIdentifier` element is instantiated in descriptions compliant to the TVA Metadata Specification, it shall be set to the value “CRID“
- The `ProgramIdentifier` element instances in descriptions compliant to the TVA Metadata Specification shall specify a CRID that complies with the syntax defined in the TVA Content Referencing Specification.

5.4.1.5 Informative Examples

Informative examples of the usage history description schemes presented in this section are provided in section 15.2 of [2].

5.4.1.6 Related MPEG-7 types (informative)

This section contains a list of types in MPEG-7 related to the usage history description schemes. The following MPEG-7 types are used indirectly by the usage history description schemes.

- DSType – see [2], section 5.4
- Mpeg7Type - see [2], section 5.4
- HeaderType - see [2], section 5.4
- ReferenceType - see [2], section 6.1.1
- UniqueIDType – see [2], section 6.2
- TermType - see [2], section 7.3.2
- TimeType - see [2], section 6.3.1
- timePointType - see [2], section 6.3.2
- durationType - see [2], section 6.3.3
- IncrDurationType - see [2], section 6.3.4
- RelTimePointType - see [2], section 6.3.5

- RelIncrTimePointType - see [2], section 6.3.6
- MediaTimeType - see [2], section 6.3.8
- mediaTimePointType - see [2], section 6.3.9
- mediaDurationType - see [2], section 6.3.10
- MediaIncrDurationType - see [2], section 6.3.11
- MediaRelTimePointType - see [2], section 6.3.12
- MediaRelIncrTimePointType - see [2], section 6.3.13
- xPathType - see [2], section 6.1.1
- TextualType - see [2], section 7.2.1
- protectionType - see [2], section 15.1.1
- ControlledTermUseType - see [2], section 7.3.3
- termRelationKindType - see [2], section 7.3.3
- controlledTermIdentifierType - see [2], section 7.3.5
- classificationSchemeIdentifierType - see [2], section 7.3.6
- classificationSchemeLocatorType - see [2], section 7.3.7
- UserIdentifierType - see [2], section 15.1.3

5.4.2 User Preferences DS

This section contains description schemes that facilitate description of user's preferences pertaining to consumption of multimedia material. User preference descriptions can be correlated with media descriptions to search, filter, select and consume desired content. Correspondence between user preference and media descriptions facilitates accurate and efficient personalization of content access and content consumption.

In particular, usage scenarios enabled by these schemes include the following:

- Identification of multiple users
- Filtering according to a rich combination of user preferences on genre, time, date, channel, etc.
- Accurate and effective agent operation by featuring a well-defined mapping between user preferences and media descriptions
- Prioritization of sources of information in combination with other preferences such as genres, titles, etc.
- Specification of preferences (e.g., for a favorite actor) for a particular time duration
- Specification of preferred keywords in connection with other preferences, such as genre (e.g., news).
- Specification of preferred critics and critic’s ratings
- Description of consumer’s desire to keep the entire, or selected parts of preference data private
- Specification of preferences for genre and source preference combinations.
- Descriptions of preferences for particular kinds of highlights (e.g., highlights of certain durations or highlights composed of segments containing certain events).
- Exchange of personal profiles under consumer control
- Specification of profiles for different countries.

The TV-Anytime Forum UserPreferences schema is based on the UserPreferences description scheme (DS) as defined in [2], section 15. [4]

An entity-relation diagram of the entities of the UserPreferences description scheme is shown in Figure 10.

The UserPreferences DS is associated with a particular user (or group of users) by means of the UserIdentifier DS. The main entity in the diagram, the UsagePreferences DS, contains two main components, the BrowsingPreferences DS and the FilteringAndSearchPreferences DS. The BrowsingPreferences DS can be used to specify preferences on the way the content is consumed, and contains SummaryPreferences. The FilteringAndSearchPreferences DS can be used to specify preferences on the type of content to be searched, filtered, selected and consumed. This DS contains the ClassificationPreferences DS, CreationPreferences DS and SourcePreferences DS.

The UserPreferences DS enables users to specify preferences that apply only in a particular context, in terms of time and place, using the PreferenceCondition DS. The UserPreferences DS allows users to specify the relative importance of their preferences with respect to each other. The DS enables users to indicate whether their preferences or parts of their preferences should be kept private or not. The DS also enables users to indicate whether the automatic update of their usage preferences description, e.g., by an agent, should be permitted or not. The ClassificationPreferences DS is used to specify user preferences related to classification of the content, e.g., preferred genre, preferred country of origin or preferred language. The CreationPreferences DS is used to specify user's preferences related to the creation description of AV content, such as preference on a particular title, or a favorite actor, or period of time within which the content was created. The SourcePreferences DS is used to specify preferences for the source of the media, such as its medium, or its distributor or publisher.

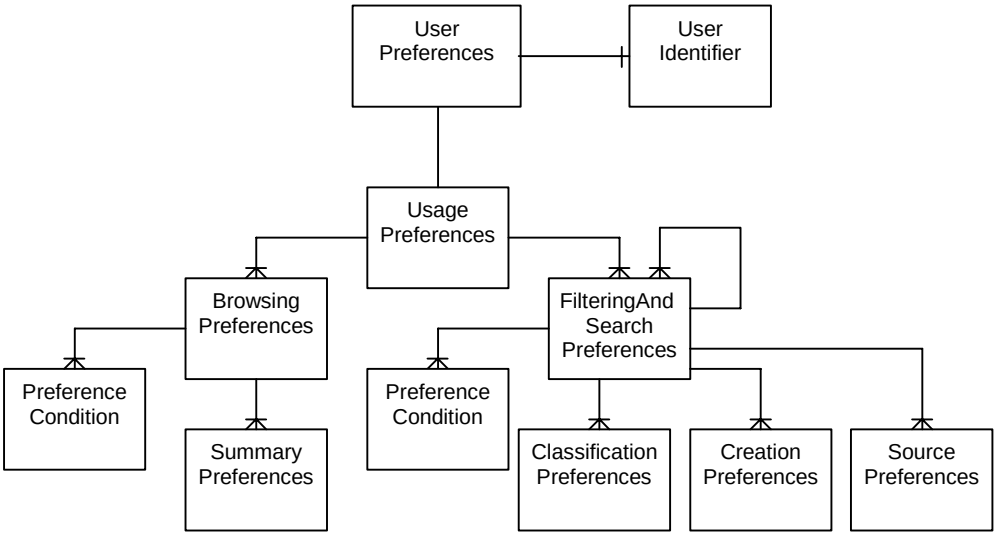

Figure 10 User preferences description schemes

The following sections contain a specification of the syntax and semantics of the UserPreferences description scheme.

[4] These sections are taken from the current MPEG-7 MDS Final Committee Draft [2], where the User Preferences DS specification is located. It is expected that a future version of this document will refer to the MPEG-7 MDS International Standard.

5.4.2.1 Basic User Preference Elements

The PreferenceCondition DS is used to specify a combination of time and/or place to be associated with a particular set of user preferences. The protectionType datatype specifies whether the user allows others access to his/her preference descriptions. The allowAutomaticUpdateType datatype signals whether the user permits his/her usage preferences to be updated automatically. The preferenceValue attribute describes the relative significance of a particular preference element.

The specifications of these basic user preference elements are given in section 15.1.1 of [2]. Syntax of the basic user preference elements is specified in section 15.1.1.1 of [2]. Semantics of the basic user preference elements are specified in section 15.1.1.2 of [2].

5.4.2.2 UserPreferences DS

The UserPreferences DS is used to describe a user's preferences for consumption of multimedia material. Correspondence between user preference information and media descriptions allows personalization of content access and content consumption.

The specification of the UserPreferences DS is given in section 15.1.2 of [2]. Syntax of the UserPreferences DS is described in section 15.1.2.1 of [2]. Semantics of the UserPreferences DS is described in section 15.1.2.2 of [2].

5.4.2.3 UserIdentifier DS

The UserIdentifier DS associated a specific user (or set of users) with a particular user preference description.

The specification of the UserPreference DS is given in section 15.1.3 of [2]. Syntax of the UserIdentifier DS is specified in section 15.1.3.1 in [2]. Semantics of the UserIdentifier DS are specified in section 15.1.3.2 in [2].

5.4.2.4 UsagePreferences DS

The UsagePreferences DS is used to specify user's preferences for filtering, searching (e.g. favorite titles, genre and actors) and browsing (e.g. preferred views of favorite programs) AV content.

The specification of the UsagePreferences DS is given in section 15.1.4 of [2]. Syntax of the UsagePreferences DS is specified in section 15.1.4.1 in [2]. Semantics of the UsagePreferences DS are specified in section 15.1.4.2 in [2].

5.4.2.5 FilteringAndSearchPreferences DS

The FilteringAndSearchPreferences DS specifies a user's filtering and/or searching preferences for audio-visual content. These preferences can be specified in terms of creation-, classification- and source-related properties of the content.

The specification of the FilteringAndSearchPreferences DS is given in section 15.1.5 of [2]. Syntax of the FilteringAndSearchPreferences DS is specified in section 15.1.5.1 in [2]. Semantics of the FilteringAndSearchPreferences DS are specified in section 15.1.5.2 in [2].

5.4.2.6 CreationPreferences DS

The CreationPreferences DS specifies a user's preferences about the creation-related properties of AV content, such as favorite actors etc.

The specification of the CreationPreferences DS is given in section 15.1.6 of [2]. Syntax of the CreationPreferences DS is specified in section 15.1.6.1 in [2]. Semantics of the CreationPreferences DS are specified in section 15.1.6.2 in [2].

5.4.2.7 ClassificationPreferences DS

The ClassificationPreferences DS is used to convey a user's preferences about various classifications of the content, such as preferred genre or language.

The specification of the ClassificationPreferences DS is given in section 15.1.7 of [2]. Syntax of the ClassificationPreferences DS is specified in section 15.1.7.1 in [2]. Semantics of the ClassificationPreferences DS are specified in section 15.1.7.2 in [2].

5.4.2.8 SourcePreferences DS

The SourcePreferences DS is used to convey preferences on the source of the AV content, such as publisher or channel of distribution. The specification of the SourcePreferences DS is given in section 15.1.8 of [2]. Syntax of the SourcePreferences DS is specified in section 15.1.8.1 in [2]. Semantics of the SourcePreferences DS are specified in section 15.1.8.2 in [2].

5.4.2.9 BrowsingPreferences DS

The BrowsingPreferences DS is used to specify a user's preferences for navigating and accessing multimedia content.

The specification of the BrowsingPreferences DS is given in section 15.1.9 of [2]. Syntax of the BrowsingPreferences DS is specified in section 15.1.9.1 in [2]. Semantics of the BrowsingPreferences DS are specified in section 15.1.9.2 in [2].

5.4.2.10 SummaryPreferences DS

The SummaryPreferences DS describes a user's preferences for nonlinear navigation of media especially with respect to visualization and sonification of AV content.

The specification of the SummaryPreferences DS is given in section 15.1.10 of [2]. Syntax of the BrowsingPreferences DS is specified in section 15.1.10.1 in [2]. Semantics of the BrowsingPreferences DS are specified in section 15.1.10.2 in [2].

5.4.2.11 Related MPEG-7 types (informative)

This section contains a list of types in MPEG-7 related to the user preference description schemes. The following MPEG-7 types are used indirectly by the user preference description scheme.

- DSType – see [2], section 5.4
- Mpeg7Type - see [2], section 5.4
- HeaderType - see [2], section 5.4
- TimeType - see [2], section 6.3.1
- timePointType - see [2], section 6.3.2
- durationType - see [2], section 6.3.3
- IncrDurationType - see [2], section 6.3.4
- RelTimePointType - see [2], section 6.3.5
- RelIncrTimePointType - see [2], section 6.3.6
- mediaDurationType - see [2], section 6.3.10
- ReferenceType - see [2], section 6.1.1
- xPathType - see [2], section 6.1.1
- TextualType - see [2], section 7.2.1
- TermType - see [2], section 7.3.2

- ControlledTermUseType - see [2], section 7.3.3
- termRelationKindType - see [2], section 7.3.3
- controlledTermIdentifierType - see [2], section 7.3.5
- classificationSchemeIdentifierType - see [2], section 7.3.6
- classificationSchemeLocatorType - see [2], section 7.3.7
- CreatorType - see [2], section 9.1.2
- PersonType - see [2], section 7.4.2
- PersonNameType - see [2], section 7.4.5
- NameComponentType - see [2], section 7.4.5
- AgentType - see [2], section 7.4.1
- AgentRefType - see [2], section 7.4.1
- PersonGroupType - see [2], section 7.4.3
- PersonGroupRefType - see [2], section 7.4.3
- OrganizationType - see [2], section 7.4.4
- OrganizationRefType - see [2], section 7.4.4
- ElectronicAddressType - see [2], section 7.4.6
- CreationMaterialType - see [2], section 9.1.2
- PlaceType - see [2], section 7.5.1
- PlaceRefType - see [2], section 7.5.1
- LanguageType - see [2], section 9.1.3
- GenreType - see [2], section 9.1.3
- ParentalGuidanceType - see [2], section 9.1.3
- MediaFormatType - see [2], section 8.1.4
- TitleType - see [2], section 9.1.2
- MediaReviewType - see [2], section 9.1.3
- summaryComponentType - see [2], section 13.1.3

5.5 Segmentation Metadata

Segmentation refers to the ability to define, access and manipulate temporal intervals (*i.e.*, segments) within an AV stream. By associating metadata with segments and segment groups, it is possible to restructure and re-purpose an input AV stream to generate alternative consumption and navigation modes. Such modes could include, for example, a summary of the content with highlights, or a set of bookmarks that point to "topic headings" within the stream. Such metadata can be provided by service providers or broadcasters as a value-added feature, and/or generated by viewers themselves. Applications include, for example, repurposing of content for educational purposes.

5.5.1 Segmentation Metadata: Definitions and Requirements

In this section we present an overview of segmentation, including definitions of terminology and a list of requirements for common applications.

An entity-relationship diagram of the various components of a (segmented) program is shown in Figure 11. The properties and relationships of each entity are provided in more detail in the following paragraphs.

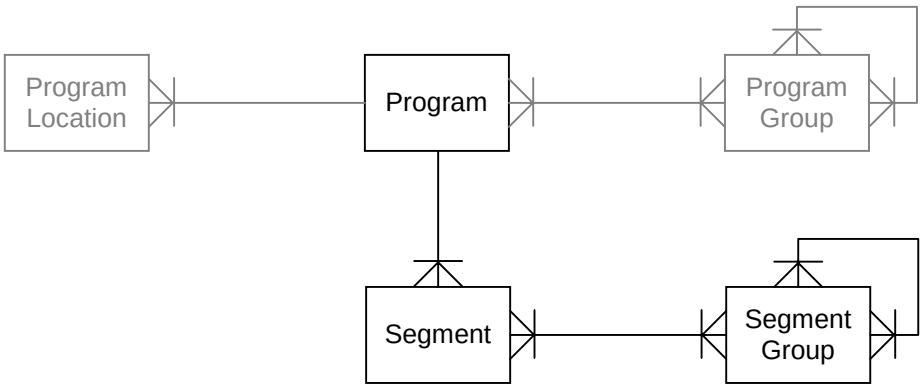

Figure 11 Entity-relationship graph for the segmentation-related components of a TVA system

Entity Definitions:

Program – the program represents an editorially coherent piece of content unambiguously identified by a CRID.

Program Group – the program group entity defines a grouping of programs. Program groups can also contain other program groups. The relevant group type for segmentation applications is the 'Program Compilation' group type, which allows Segments from multiple programs to be combined in Segment Groups.

Program Location – Program Location provides a physical location where the program is available. A program may be available at multiple program locations; selection of a particular program location is performed during the location resolution process. The timelines of different instances of a program identified by a given CRID are assumed to be identical; hence it is inconsequential for the segmentation description which location is selected during the resolution process.

Segment – A segment is a continuous fragment of a program. A particular segment can belong to a single program only, but it can be a member of multiple segment groups.

Segment Group – denotes a collection of segments that are grouped together, for a particular purpose or due to a shared property. A segment group can contain segments, or other segment groups.

Relationship Definitions:

Program-to-Segment: A Segment is part of a single program, which is identified by its CRID. A Program can contain multiple segments.

Segment-to-Segment Group: A Segment can belong to zero or more Segment Groups. A Segment Group can contain zero or more Segments (possibly from multiple Programs).

Segment Group-to-Segment Group: A Segment Group can be a member of zero or more Segment Groups, and it can contain zero or more Segment Groups. A Segment Group may contain either segments, or subgroups, but not both (*This latter constraint is imposed by the syntax and semantics of the description schemes*).

5.5.2 Basic Segment Description

The following complex type defines descriptive properties of segments.

```
<complexType name="BasicSegmentDescriptionType">
  <sequence>
    <element name="Title" type="mpeg7:TitleType"
    minOccurs="0" maxOccurs="unbounded"/>
    <element name="Synopsis" type="tva:SynopsisType"
    minOccurs="0" maxOccurs="unbounded"/>
    <element name="Keywords" type="mpeg7:KeywordAnnotationType"
    minOccurs="0" maxOccurs="1"/>
    <element name="RelatedMaterial" type="tva:RelatedMaterialType”
    minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>
```

Name	Definition
BasicSegmentDescriptionType	Defines basic segment description
Title	A title of the segment (optional). A segment can have several titles when necessary, e.g. in different languages.
Synopsis	A synopsis or textual description of the segment (optional). A segment can have several synopses when necessary, e.g. in different languages or lengths.
Keywords	A list of keywords associated with the segment (optional). A segment can have several keywords when necessary, e.g. in different languages.
RelatedMaterial	A link to external material related to the segment (optional). A segment can have multiple links.

5.5.3 Segment Information

The following element and complex type define a segment.

```
<element name="SegmentInformation"
  type="tva:SegmentInformationType"/>

<complexType name="SegmentInformationType">
  <sequence>
    <element name="ProgramRef" type="tva:CRIDRefType" minOccurs="0"/>
    <element name="Description"
    type="tva:BasicSegmentDescriptionType" minOccurs="0"/>
    <element name="SegmentLocator" type="mpeg7:MediaTimeType"/>
    <element name="KeyFrameLocator" type="mpeg7:MediaTimeType"
    minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="segmentId" type="ID" use="required"/>
  <attribute name="version" type="integer" use="optional"/>
</complexType>
```

Name	Definition
SegmentInformation	An element that instantiates the SegmentInformationType.
SegmentInformationType	Defines an individual segment.
ProgramRef	A reference to the program this segment belongs to. When the ProgramRef element is not instantiated within a segment, the program that the segment belongs to is specified by the ProgramRef element of (one of) its parent segment group(s). When the segment is a direct member of a segment group that defines a program compilation (i.e., the ProgramRef element of the parent segment group references a CRID associated with a Program Compilation), the ProgramRef element of the segment will reference the CRID of the particular program that the segment belongs to
Description	A description of the content of the segment.
SegmentLocator	Locates the segment within a program (instance) in terms of start time and duration (optional). Defined as an MPEG-7 datatype, MediaTimeType (See Sec. 6.3.9 of [2] for a detailed description). If the duration is not specified, the segment ends at the end of the program. If the timeBase attribute for the SegmentLocator element is not present, the time base for the segment is taken to be the start point of the program identified by the associated CRID. If the timeUnit attribute for the SegmentLocator element is not present, the default time unit provided with the SegmentInformationTable description is adopted. If the timeBase/timeUnit attributes of the SegmentLocator element are present, their values override the defaults provided in the SegmentInformationTable.
KeyFrameLocator	Locates a key frame of the segment within a program in terms of a time point (optional). Defined as an MPEG-7 datatype, MediaTimeType (See Sec. 6.3.9 of [2] for a detailed description). MediaDuration and MediaIncrDuration elements of a KeyFrameLocator element shall not be used. Multiple key frames may be associated with a single segment. If the timeBase and/or timeUnit attributes for the KeyFrameLocator element are instantiated, they override the default or global time base and time unit definitions provided with the SegmentInformationTable description.
segmentId	The unique identifier of the segment.
version	The version number of the segment information (optional).

5.5.4 Segment Group Information

The following element and complex types define segment grouping.

```
<element name="SegmentGroupInformation"
  type="tva:SegmentGroupInformationType"/>

<complexType name="SegmentGroupInformationType">
  <sequence>
    <element name="ProgramRef" type="tva:CRIDRefType"/>
    <element name="Description"
    type="tva:BasicSegmentDescriptionType" minOccurs="0"
    maxOccurs=”1”/>
    <element name="GroupInterval" minOccurs="0" maxOccurs=”1”>
    <complexType>
      <attribute name="ref" type="IDREF"/>
    </complexType>
    </element>
    <choice minOccurs="0" maxOccurs="1">
    <element name="Segments">
      <complexType>
        <attribute name="refList" type="IDREFS" use=”required”/>
      </complexType>
    </element>
    <element name="Groups">
      <complexType>
        <attribute name="refList" type="IDREFS" use=”required”/>
      </complexType>
    </element>
    </choice>
    <element name="KeyFrameLocator" type="mpeg7:MediaTimeType"
    minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="groupId" type="ID" use="required"/>
  <attribute name="groupType" type="tva:SegmentGroupTypeType"
    use="required"/>
  <attribute name="ordered" type="boolean" use="optional"/>
  <attribute name="numberOfSegments" type="positiveInteger"
    use="optional"/>
  <attribute name="numberOfKeyFrames" type="positiveInteger"
    use="optional"/>
  <attribute name="duration" type="mpeg7:mediaDurationType"
    use="optional"/>
  <attribute name="topLevel" type="boolean" use="optional"/>
  <attribute name="version" type="integer" use="optional"/>
</complexType>

<simpleType name="SegmentGroupTypeType">
  <list>
    <simpleType>
      <restriction base="string">
        <enumeration value="highlights"/>
        <enumeration value="highlights/objects"/>
        <enumeration value="highlights/events"/>
        <enumeration value="bookmarks"/>
        <enumeration value="bookmarks/objects"/>
        <enumeration value="bookmarks/events"/>
        <enumeration value="themeGroup"/>
```

```
      <enumeration value="preview"/>
      <enumeration value="preview/title"/>
      <enumeration value="preview/slideshow"/>
      <enumeration value="tableOfContents"/>
      <enumeration value="synopsis"/>
      <enumeration value="shots"/>
      <enumeration value=”insertionPoints”/>
      <enumeration value="alternativeGroups"/>
      <enumeration value="other"/>
    </restriction>
  </simpleType>
 </list>
</simpleType>
```

Name	Definition
SegmentGroupInformation	An element that instantiates the SegmentGroupInformationType
SegmentGroupInformationType	Defines an individual segment group
ProgramRef	A reference to the program this segment group belongs to. When the member segments/groups are collected from different programs, the ProgramRef element references the CRID of a program group of type “programCompilation.” This CRID is resolved into the individual programs CRIDs.
Description	A description of the content of the segment group
GroupInterval	References a single segment that defines the temporal range of the segment group.
Segments	Defines the segments that are part of this group by providing a list of references to the identifiers of elements of type SegmentInformationType (optional). The order of the references to segments in this list determines the temporal playback order of segments within this group.
Groups	Defines the segment groups that are subgroups of this group by providing a list of references to the identifiers of elements of type SegmentGroupInformationType (optional). The order of the references to segment groups in this list determines their ordering within this group.
KeyFrameLocator	Locates a key frame of the segment group within a program in terms of a time point (optional). Defined as an MPEG-7 datatype, MediaTimeType (See Sec. 6.3.9 of [2] for a detailed description). MediaDuration and MediaIncrDuration elements of a KeyFrameLocator element shall not be used. Multiple key frames may be associated with a single segment group.
groupId	The unique identifier of the segment group
groupType	The type of the segment group. A list of the valid segment group types is defined by

Name	Definition
	SegmentGroupTypeType.
ordered	Specifies whether the given segment group presents an ordered list (i.e. whether order of the segment or segment groups within the given segment group is significant) (optional).
numberOfSegments	The number of segments in the segment group (optional). The value of this attribute specifies only the segments that are direct members of the segment group.
numberOfKeyFrames	The number of key frames in the segment group (optional). The value of this attribute specifies only the key frames of the segments that are direct members of the segment group.
duration	The sum of the durations of the segments contained within this group (optional). This duration corresponds to the sum of the durations of the segments that are direct members of the segment group.
topLevel	Specifies whether the given segment group is a top-level group (optional).
version	The version number of the segment group (optional).

Name	Definition
SegmentGroupTypeType	A simple type that specifies the valid types of segment groups. The allowed types are defined as follows: *highlights* - The group of segments represents selected highlights from one or more programs. A segment group of this type requires continuous playback. *highlights/objects* - The group of segments represents selected highlights from a program (or programs) that share a common object or objects (e.g. *Seinfeld* highlights with Kramer). A segment group of this type requires continuous playback. *highlights/events* - The group of segments represents selected highlights from a program (or programs) that share a common event or events (e.g. touchdowns in the Super Bowl). A segment group of this type requires continuous playback. *bookmarks* - The segment group defines a set of access points to a program. If the member segments of a segment group of type *bookmarks* contain segment duration information, this duration information shall be ignored, and the segments shall be treated as "open-ended." A segment group of this type

Name	Definition
	does not require continuous playback. *bookmarks/objects* - The segment group defines a set of access points to a program, where the selected access points share a common object or objects. If the member segments of a segment group of type *bookmarks/objects* contain segment duration information, this duration information shall be ignored, and the segments shall be treated as “open-ended.” A segment group of this type does not require continuous playback. *bookmarks/events* - The segment group defines a set of access points to a program, where the selected access points share a common event or events. If the member segments of a segment group of type *bookmarks/events* contain segment duration information, this duration information shall be ignored, and the segments shall be treated as “open-ended.” A segment group of this type does not require continuous playback. *themeGroup* – The segment group comprises segments that share a common topic or theme. The common theme can be specified in the segment group description. A theme group does not necessarily require direct continuous playback. *preview* - The segment group defines a preview of a program. A segment group of this type requires continuous playback. *preview/title* - The segment group defines a preview of a program, where the preview serves as a promotional title or trailer for the program. A segment group of this type requires continuous playback. *preview/slideshow* - The segment group defines a preview of a program, where the preview serves as a compact slideshow of the program content. A segment group of this type requires continuous playback. *tableOfContents* - The segment group defines a navigable table of contents for the program. A segment group of this type does not require continuous playback. *synopsis* - The segment group provides a summary or synopsis of the program. A segment group of this type requires continuous playback. *shots* - The segment group provides a list of the shots in the program. A segment group of this type does not require continuous playback. *insertionPoints* – The segment group provides a list of segments which function as insertion points into the program of interest; e.g. temporal

Name	Definition
	locations of the commercials to be shown during a program. The duration information associated with member segments in a segment group of type `insertionPoints` is ignored, since the member segments only determine the time instances in the original program where additional content is to be inserted. A segment group of this type does not require continuous playback. *alternativeGroups* - Each member of this type of segment group provides an alternative view or representation, with the same functionality but different durations or levels of detail. A segment group of this type does not require continuous playback. *other* – any other segment group type.

Validity constraints:

Various validity constraints are imposed on the proposed description scheme to ensure that (i) it fits the data model of Figure 1, and (ii) the sequence and relationships of the various segments and segment groups are unambiguously defined. These constraints, which are implicit in the description schemes, are outlined below also for clarity:

- A segment group may contain either segments, or subgroups, but not both.
- A segment group of type "`alternativeGroups`" may not contain segments and shall only contain subgroups.
- A segment group of any type other than "`tableOfContents`" and "`alternativeGroups`" may only contain segments. A group of type "`tableOfContents`" may contain other segment groups of type "`tableOfContents`."

These validity constraints reduce the complexity of the resulting descriptions by limiting the degree of nesting in the hierarchy. The navigation order of segments or segment groups is determined by the order of references to the segments in a segment group.

5.5.5 Segment Information Table

The following element and complex type define a structure for holding all segmentation-related metadata.

```
<element name="SegmentInformationTable"
 type="tva:SegmentInformationTableType"/>

<complexType name="SegmentInformationTableType">
  <sequence>
    <element name="SegmentList">
      <complexType>
        <sequence>
          <element ref="tva:SegmentInformation" minOccurs="1"
            maxOccurs="unbounded"/>
        </sequence>
      </complexType>
    </element>
```

```
    <element name="SegmentGroupList">
      <complexType>
        <sequence>
          <element ref="tva:SegmentGroupInformation" minOccurs="0"
            maxOccurs="unbounded"/>
        </sequence>
      </complexType>
    </element>
    <element name="TimeBaseReference" minOccurs="0"
      maxOccurs="unbounded">
      <complexType>
        <choice>
          <element name="RefMediaTime"
            type="mpeg7:mediaTimePointType"/>
          <element name="RefURI" type="anyURI"/>
        </choice>
      </complexType>
    </element>
  </sequence>
  <attribute name="timeUnit" type="mpeg7:mediaDurationType"
    use="optional" default="PT1N1000F"/>
  <attribute name="version" type="integer" use="optional"/>
</complexType>
```

Name	Definition
`SegmentInformationTable`	An element that instantiates the `SegmentInformationTableType`
`SegmentInformationTableType`	Defines a structure for holding all segmentation-related metadata
`SegmentList`	The list of the segments in the `SegmentInformationTable`
`SegmentGroupList`	The list of the segment groups in the `SegmentInformationTable`
`TimeBaseReference`	Defines the time base reference(s) for the current description. Multiple time base references can be specified for a single `SegmentInformationTable`. These references can be referred to by the `timeBase` attributes of the `SegmentLocator` and `KeyFrameLocator` elements of the description. If no `TimeBaseReference` is provided in the description, the time base is taken to be the start point of the program identified by the associated CRID.
`RefMediaTime`	Specifies the time base reference using an element of MPEG-7 type `mediaTimePointType` (See Sec. 6.3.10 of [2] for a detailed description)
`RefURI`	Specifies the time base reference using an element of type `anyURI`
`timeUnit`	Specifies the duration of the time intervals used in the incremental specifications of relative time points and duration. Default time unit is milliseconds.

`version`	The version number of the segment information table (optional)

5.6 TV Anytime documents

TV-Anytime metadata is structured into self-contained documents. Each document has a single top-level element that encloses all other TV-Anytime metadata, as described below.

5.6.1 Information tables

```
<element name="ProgramInformationTable">
  <complexType>
    <sequence>
      <element ref="tva:ProgramInformation" minOccurs="0"
        maxOccurs="unbounded"/>
    </sequence>
    <attribute name="version" type="integer"/>
  </complexType>
</element>

<element name="GroupInformationTable">
  <complexType>
    <sequence>
      <element ref="tva:GroupInformation" minOccurs="0"
        maxOccurs="unbounded"/>
    </sequence>
    <attribute name="version" type="integer"/>
  </complexType>
</element>

<element name="ProgramLocationTable">
  <complexType>
    <sequence>
      <choice minOccurs="0" maxOccurs="unbounded">
        <element name="OnDemandProgram"
          type="OnDemandProgramPublicationType"/>
        <element name="BroadcastEvent" type="BroadcastEventType"/>
        <element name="Schedule" type="ScheduleType"/>
      </choice>
    </sequence>
    <attribute name="version" type="integer"/>
  </complexType>
</element>

<element name="ServiceInformationTable">
  <complexType>
    <sequence>
      <element ref="tva:ServiceInformation" minOccurs="0"
        maxOccurs="unbounded"/>
    </sequence>
    <attribute name="version" type="integer"/>
  </complexType>
</element>

<element name="CreditsInformationTable">
  <complexType>
    <sequence>
      <element name="Agent" type="mpeg7:AgentType" minOccurs="0"
        maxOccurs="unbounded"/>
    </sequence>
```

```
    <attribute name="version" type="integer"/>
  </complexType>
</element>
```

Name	Definition
ProgramInformationTable	An element that contains a table of program information records
ProgramInformation	A list of program information records
version	A version number for the table
GroupInformationTable	An element that contains a table of group information records
GroupInformation	A list of group information records
version	A version number for the table
ProgramLocationTable	An element that contains a table of program location records
ProgramLocation	A list of program location records
version	A version number for the table
ServiceInformationTable	An element that contains a table of service information records
ServiceInformation	A list of service information records
version	A version number for the table
CreditsInformationTable	An element that contains the credits information for the content. Each member of the CreditsInformationTable can be referenced by the AgentRef element of the CreditsItem element in BasicProgramDescriptionType.
Agent	A member element in the list of credits
version	A version number for the table

5.6.2 TV Anytime program information document

```
<!-- 5.6.2 TV Anytime program information document    -->
<!-- ################################################ -->
<!-- Definition of TVAMain DS                         -->
<!-- ################################################ -->

<element name="TVAMain" type="tva:TVAMainType"/>

<complexType name="TVAMainType">
  <sequence>
    <element name="CopyrightNotice" type="string" minOccurs="0"/>
    <element ref="ClassificationSchemeTable" minOccurs="0"/>
    <element ref="tva:ContentDescription" minOccurs="0"
      maxOccurs="unbounded"/>
    <element ref="tva:UserDescription" minOccurs="0"
      maxOccurs="unbounded"/>
  </sequence>
```

```
  <attribute name="version" type="integer"/>
  <attribute ref="xml:lang" default="en"/>
  <attribute name="publisher" type="string" use="optional"/>
  <attribute name="publicationTime" type="dateTime" use="optional"/>
  <attribute name="rightsOwner" type="string" use="optional"/>
</complexType>

<element name="ProgramDescription"
  type="tva:ProgramDescriptionType"/>
<element name="UserPreferences" type="mpeg7:UserPreferencesType"/>
<element name="UsageHistory" type="mpeg7:UsageHistoryType"/>
<element name="ClassificationSchemeTable"
  type="tva:ClassificationSchemeTableType"/>

<element name="UserDescription">
  <complexType>
    <sequence>
      <element ref="tva:UserPreferences" minOccurs="0"/>
      <element ref="tva:UsageHistory" minOccurs="0"/>
    </sequence>
  </complexType>
</element>

<element name="ContentDescription">
  <complexType>
    <sequence>
      <element ref="tva:ProgramDescription" minOccurs="0"/>
    </sequence>
  </complexType>
</element>

<complexType name="ClassificationSchemeTableType">
  <sequence>
    <element name="CSAlias"
      type="mpeg7:ClassificationSchemeAliasType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="ClassificationScheme"
      type="mpeg7:ClassificationSchemeType" minOccurs="0"
      maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="ProgramDescriptionType">
  <sequence>
    <element ref="tva:ProgramInformationTable" minOccurs="0"/>
    <element ref="tva:GroupInformationTable" minOccurs="0"/>
    <element ref="tva:ProgramLocationTable" minOccurs="0"/>
    <element ref="tva:ServiceInformationTable" minOccurs="0"/>
    <element ref="tva:CreditsInformationTable" minOccurs="0"/>
    <element ref="tva:ProgramReviewTable" minOccurs="0"/>
    <element ref="tva:SegmentInformationTable" minOccurs="0"/>
  </sequence>
</complexType>
```

Name	Definition
TVAMain	The root element for a TVA schema valid instance document that provides a complete description

TVAMainType	Specifies the root element for a TVA schema valid instance document that provides a complete description
CopyrightNotice	Specifies the copyright information for the description
ClassificationSchemeTable	Contains the classification schemes used by the various descriptions in the TVA document and their aliases (optional)
ContentDescription	Contains elements for description of AV content
UserDescription	Contains elements for description of a user's preferences or content consumption history
version	Specified the version of the description
xml:lang	Specifies the language of the description. Default is 'English.'
publisher	Specifies the name of the publisher of the description
publicationTime	Specifies the time the metadata description was published.
rightsOwner	Specifies the entity that holds the rights to the description
ClassificationSchemeTableType	A complex data type for listing the classification schemes used by the various descriptions in the TVA document and their aliases
CSAlias	Specifies an alias for a ClassificationScheme referenced by a URI (optional). Defined as MPEG-7 type ClassificationSchemeAliasType (See Sec. 7.3.6 of [2] for a detailed description).
ClassificationScheme	Specifies a complete classification scheme that is transmitted as part of the TVA description document (optional). Defined as MPEG-7 type ClassificationSchemeType (See Sec. 7.3.1 of [2] for a detailed description).
ProgramDescriptionType	A complex type that aggregates the tables that contain program description metadata
ProgramInformationTable	The program information table
GroupInformationTable	The group information table
ProgramLocationTable	The program location table
ServiceInformationTable	The service information table
CreditsInformationTable	The credits information table
ProgramReviewTable	The program review table
SegmentInformationTable	The segment information table

6. The TVA Metadata Framework (Informative)

There are different contexts where it might be preferred to continuously use XML in its textual form. Binarization allows saving bandwidth and maximizing the performance of the system (parsing at the binary level is more efficient). The transport and extraction of textual or binarised XML it out of the scope of TV-Anytime, which is agnostic to the delivery means. The choice of XML as a representation format is still compatible with the delivery of metadata originated in another format.

Figure 12 shows how the system can be fed by different metadata descriptions encoded in different XML or non-XML formats.

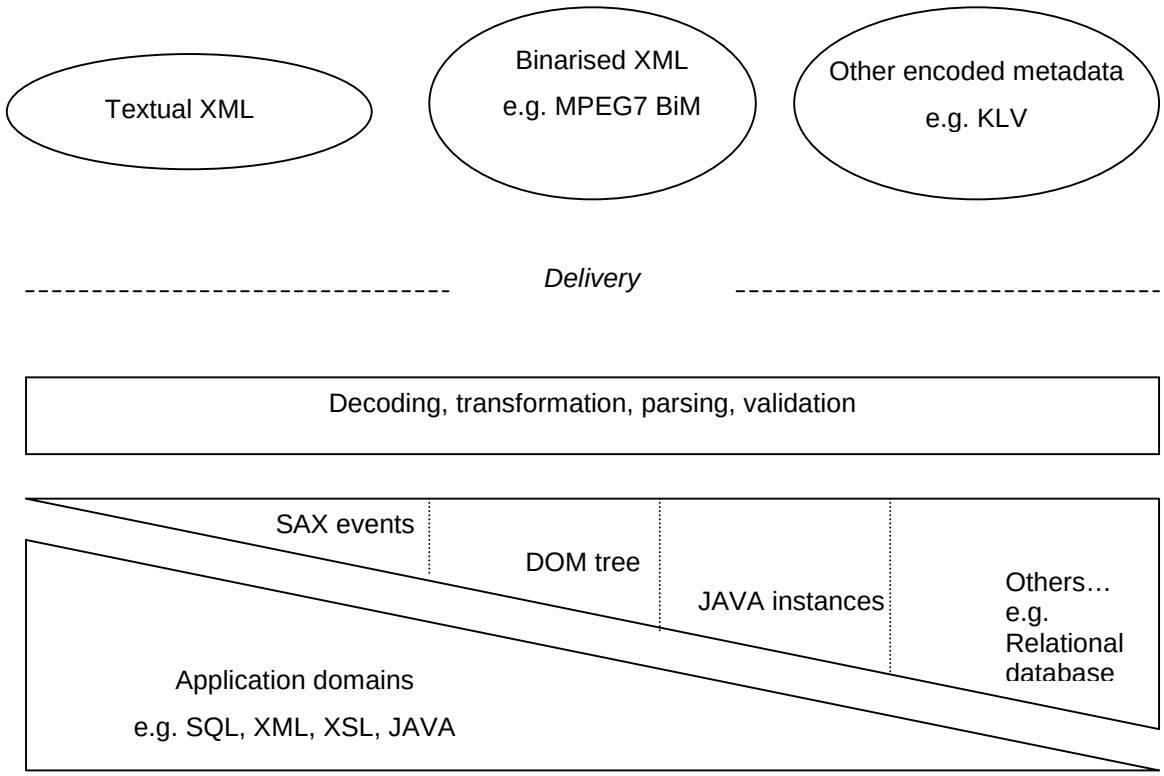

Figure 12 The XML-based TVA metadata framework

After delivery, metadata descriptions can be decoded, transformed, parsed and validated, as necessary. TV-Anytime does not specify how this should be implemented. But this is the gateway to different application domains that allows TV-Anytime to be platform independent.

7. References

[1] XML Schema, W3C Recommendations (version 20010502), available at:

http://www.w3.org/TR/2001/REC-xmlschema-0-20010502

http://www.w3.org/TR/2001/REC-xmlschema-1-20010502

http://www.w3.org/TR/2001/REC-xmlschema-2-20010502

[2] Text of ISO/IEC FCD 15938-5 Information Technology - Multimedia content description interface - Part 5 Multimedia Description Schemes, N3966, Singapore, March 2001. Available at: http://www.cselt.it/mpeg/working_documents.htm.

[3] Text of ISO/IEC FCD 15938-2 Information Technology - Multimedia content description interface - Part 2 Description Definition Language, N4002, Singapore, March 2001. Available at: http://www.cselt.it/mpeg/working_documents.htm.

Appendix A TV-Anytime Usage History Thesaurus

The following table contains a thesaurus for valid terms of the ActionType element in the Usage History Description Scheme. Terms can be referred to either by a numeric identifier or by a textual label.

Term	Label	Description
1	**Audio-Video**	**Actions Related to Audio and Video**
1.1	PlayRecording	Play content from a recording
1.2	PlayStream	Play content from input stream
1.3	Record	Record input stream to local storage media
1.4	Preview	View or listen to a summary of the input stream
1.5	Pause	Pause the input stream
1.6	FastForward	Fast forward the input stream
1.7	Rewind	Rewind the input stream
1.8	SkipForward	Skip forward over a portion of the input stream
1.9	SkipBackward	Skip backward over a portion of the input stream
1.10	Mute	Turn sound off
1.11	VolumeUp	Increase volume
1.12	VolumeDown	Reduce volume
1.13	Loop/Repeat	Repeat/loop (part of) the input stream
1.14	Shuffle	Randomly select next track
1.15	SkipToEnd	Go to the beginning of the stream
1.16	SkipToStart	Go to the end of the stream
1.17	CopyCD	Copy all or part of a CD
2	**Video**	**Actions Related to Video**
2.1	Zoom	Zoom (in) to the on-screen image or sequence
2.2	SlowMotion	View input stream in slow motion
2.3	CCOn	Closed caption is on
2.4	StepForward	Advance to next frame
2.5	StepBackward	Return to previous frame
3	**Data**	**Actions Related to Miscellaneous Data**
3.1	ClickThrough	Follow an available link
3.2	ScrollUp	Scroll up in a web page/composite page
3.3	ScrollDown	Scroll down in a web page/composite page
3.4	ViewGuide	View program/resource guide
3.5	SavePage	Save web page/composite page
3.6	PrintPage	Print web page/composite page
3.7	Search	Search the web or local resources

3.8	SubmitForm	Submit a form with requested information
3.9	SubmitQuery	Submit a query
3.10	Archive	Archive content to persistent local storage media
4	**Commerce**	**Actions Related to Commerce**
4.1	Buy	Purchase a product or item
4.2	AddToWishList	Designate a product or item as possible future purchasing item
4.3	AddToCart	Designate a product or item as potential immediate purchase item

Table 1 Thesaurus for `ActionType` of Usage History

Each term has a numeric identifier, listed in the first column, and a textual label, listed in the second column. A description of each term is listed in the third column.

The following is an (informative) XML representation of the thesaurus in Appendix A. This XML representation is a valid description (fragment) with respect to the ClassificationScheme description scheme, specified in [2], section 7.3.

```
<ClassificationScheme scheme="TVAF User Action Thesaurus">
  <Item term="1">
    <Label xml:lang='en'>Audio-Video</Label>
    <Description xml:lang='en'>
      Actions Related to Audio and Video
    </Description>
    <Item term="1.1">
      <Label xml:lang='en'>PlayRecording</Label>
      <Description xml:lang='en'>
        Play content from a recording
      </Description>
    </Item>
    <Item term="1.2">
      <Label xml:lang='en'>PlayStream</Label>
      <Description xml:lang='en'>
        Play content from input stream
      </Description>
    </Item>
    <Item term="1.3">
      <Label xml:lang='en'>Record</Label>
      <Description xml:lang='en'>
        Record input stream to local storage media
      </Description>
    </Item>
    <Item term="1.4">
      <Label xml:lang='en'>Preview</Label>
      <Description xml:lang='en'>
        View or listen to a summary of the input stream
      </Description>
    </Item>
    <Item term="1.5">
      <Label xml:lang='en'>Pause</Label>
      <Description xml:lang='en'>
        Pause the input stream
      </Description>
    </Item>
    <Item term="1.6">
      <Label xml:lang='en'>FastForward</Label>
      <Description xml:lang='en'>
        Fast forward the input stream
```

```
      </Description>
    </Item>
    <Item term="1.7">
      <Label xml:lang='en'>Rewind</Label>
      <Description xml:lang='en'>
        Rewind the input stream
      </Description>
    </Item>
    <Item term="1.8">
      <Label xml:lang='en'>SkipForward</Label>
      <Description xml:lang='en'>
        Skip forward over a portion of the input stream
      </Description>
    </Item>
    <Item term="1.9">
      <Label xml:lang='en'>SkipBackward</Label>
      <Description xml:lang='en'>
        Skip backward over a portion of the input stream
      </Description>
    </Item>
    <Item term="1.10">
      <Label xml:lang='en'>Mute</Label>
      <Description xml:lang='en'>
        Turn sound off
      </Description>
    </Item>
    <Item term="1.11">
      <Label xml:lang='en'>VolumeUp</Label>
      <Description xml:lang='en'>
        Increase volume
      </Description>
    </Item>
    <Item term="1.12">
      <Label xml:lang='en'>VolumeDown</Label>
      <Description xml:lang='en'>
        Reduce volume
      </Description>
    </Item>
    <Item term="1.13">
      <Label xml:lang='en'>Repeat</Label>
      <Label xml:lang='en'>Loop</Label>
      <Description xml:lang='en'>
        Repeat/loop (part of) the input stream
      </Description>
    </Item>
    <Item term="1.14">
      <Label xml:lang='en'>Shuffle</Label>
      <Description xml:lang='en'>
        Randomly select next track
      </Description>
    </Item>
    <Item term="1.15">
      <Label xml:lang='en'>SkipToEnd</Label>
      <Description xml:lang='en'>
        Go to the beginning of the input stream
      </Description>
    </Item>
    <Item term="1.16">
      <Label xml:lang='en'>SkipForward</Label>
      <Description xml:lang='en'>
        Go to the end of the input stream
```

```
      </Description>
    </Item>
    <Item term="1.17">
      <Label xml:lang='en'>CopyCD</Label>
      <Description xml:lang='en'>
        Copy all or parts of a CD
      </Description>
    </Item>
  </Item>
  <Item term="2">
    <Label xml:lang='en'>Video</Label>
    <Description xml:lang='en'>
      Actions Related to Video
    </Description>
    <Item term="2.1">
      <Label xml:lang='en'>Zoom</Label>
      <Description xml:lang='en'>
        Zoom (in) to the on-screen image or sequence
      </Description>
    </Item>
    <Item term="2.2">
      <Label xml:lang='en'>SlowMotion</Label>
      <Description xml:lang='en'>
        View input stream in slow motion
      </Description>
    </Item>
    <Item term="2.3">
      <Label xml:lang='en'>CCOn</Label>
      <Description xml:lang='en'>
        Closed caption is on
        </Description>
    </Item>
    <Item term="2.4">
      <Label xml:lang='en'>StepForward</Label>
      <Description xml:lang='en'>
        Advance to next frame
    </Description>
    </Item>
    <Item term="2.5">
      <Label xml:lang='en'>StepBackward</Label>
      <Description xml:lang='en'>
        Return to previous frame
    </Description>
    </Item>
  </Item>
  <Item term="3">
    <Label xml:lang='en'>Data</Label>
    <Description xml:lang='en'>
      Actions Related to Miscellaneous Data
    </Description>
    <Item term="3.1">
      <Label xml:lang='en'>ClickThrough</Label>
      <Description xml:lang='en'>
        Follow an available link
      </Description>
    </Item>
    <Item term="3.2">
      <Label xml:lang='en'>ScrollUp</Label>
      <Description xml:lang='en'>
        Scroll up in a web page/composite page
      </Description>
```

```
    </Item>
    <Item term="3.3">
      <Label xml:lang='en'>ScrollDown</Label>
      <Description xml:lang='en'>
        Scroll down in a web page/composite page
      </Description>
    </Item>
    <Item term="3.4">
      <Label xml:lang='en'>ViewGuide</Label>
      <Description xml:lang='en'>
        View program/resource guide
      </Description>
    </Item>
    <Item term="3.5">
      <Label xml:lang='en'>SavePage</Label>
      <Description xml:lang='en'>
        Save web page/composite page
      </Description>
    </Item>
    <Item term="3.6">
      <Label xml:lang='en'>PrintPage</Label>
      <Description xml:lang='en'>
        Print web page/composite page
      </Description>
    </Item>
    <Item term="3.7">
      <Label xml:lang='en'>Search</Label>
      <Description xml:lang='en'>
        Search the web or local resources
      </Description>
    </Item>
    <Item term="3.8">
      <Label xml:lang='en'>SubmitForm</Label>
      <Description xml:lang='en'>
        Submit a form with requested information
      </Description>
    </Item>
    <Item term="3.9">
      <Label xml:lang='en'>SubmitQuery</Label>
      <Description xml:lang='en'>
        Submit a query
      </Description>
    </Item>
    <Item term="3.10">
      <Label xml:lang='en'>Archive</Label>
      <Description xml:lang='en'>
        Archive content to persistent local storage media
      </Description>
    </Item>
  </Item>
  <Item term="4">
    <Label xml:lang='en'>Commerce</Label>
    <Description xml:lang='en'>
      Actions Related to Commerce
    </Description>
    <Item term="4.1">
      <Label xml:lang='en'>Buy</Label>
      <Description xml:lang='en'>
        Purchase a product or item
      </Description>
    </Item>
```

```
    <Item term="4.2">
      <Label xml:lang='en'>AddToWishList</Label>
      <Description xml:lang='en'>
        Designate a product or item as possible future purchasing item
      </Description>
    </Item>
    <Item term="4.3">
      <Label xml:lang='en'>AddToCart</Label>
      <Description xml:lang='en'>
        Designate a product or item as potential
        immediate purchase item
      </Description>
    </Item>
  </Item>
</ClassificationScheme>
```

Appendix B TV-Anytime Genre Dictionary

The TV-Anytime genre dictionary is supplied as a separate document SP003-Appendix Bv11, which is attached to this document as part of a common zip file.

Appendix C TV-Anytime Description Schemes

The TV-Anytime description schemes listed in this document have been aggregated into a single file, tva11.xsd, which is included as a part of this specification and attached to this document as part of a common zip file. These description schemes, along with the MPEG-7 DSs referenced herein, constitute the normative set of TV-Anytime metadata, version 1.1.