The Application of Metadata Standards to Multimedia in Museums

Jane Hunter

DSTC Pty Ltd, The University of Queensland, Qld, 4072, Australia.
Phone +61 7 3365 4310, Fax +61 7 3365 4311 jane@dstc.edu.au

Abstract. This paper first describes the application of a multi-level indexing approach, based on Dublin Core extensions and the Resource Description Framework (RDF), to a typical museum video. The advantages and disadvantages of this approach are discussed in the context of the requirements of the proposed MPEG-7 ("Multimedia Content Description Interface") standard. The work on SMIL (Synchronized Multimedia Integration Language) by the W3C SYMM working group is then described. Suggestions for how this work can be applied to video metadata are made. Finally a hybrid approach is proposed based on the combined use of Dublin Core and the currently undefined MPEG-7 standard within the RDF which will provide a solution to the problem of satisfying widely differing user requirements.

1. Introduction

Multimedia provides museums with a communication and preservation tool capable of generating much deeper, richer interpretations of cultural artifacts than is possible through text alone. Consequently, museum multimedia databases are rapidly developing into vast storehouses of cultural knowledge and resources. However the value, accessibility and reusability of these cultural resources is largely dependent on the quality of the maps or guides to these vast, complex, multi-layered repositories. The development of content-based metadata standards for audiovisual data will provide the basis for such guides, as well as facilitate the associated multimedia capable search engines.

Dublin Core was designed specifically for generating metadata for textual documents. Although a number of workshops have been held to discuss the applicability of Dublin Core to non-textual documents such as images, sound and moving images, they have primarily focused on extensions to the 15 core elements through the use of sub-elements and schemes specific to audiovisual data, to describe bibliographic-type information rather than the actual content.

The objective of the proposed MPEG-7 ("Multimedia Content Description Interface") standard is to specify a standard set of descriptors and description schemes

for describing the content of audiovisual information. The MPEG-7 work group expects to issue a Call for Proposals in October 1998.

This paper first outlines a multi-level video indexing approach based on Dublin Core extensions and the Resource Description Framework (RDF). The advantages and disadvantages of this approach are discussed in the context of the requirements of the proposed MPEG-7 standard. The related work on SMIL (Synchronized Multimedia Integration Language) by the W3C SYMM working group is described. Suggestions for how this work can be applied to video metadata are made. Finally a hybrid approach is proposed based on the combined use of Dublin Core and the currently undefined MPEG-7 standard within the RDF which will provide a solution to the problem of satisfying widely differing user requirements.

2. Video Indexing

Detailed indexing of a film or video clip consists of the following steps:
1. Segment the video hierarchically into sequences, scenes, and shots. (A *shot* is a continuous sequence of frames captured from one camera. A *scene* is composed of one or more shots which present different views of the same event, related in time or space. A *segment* is composed of one or more related scenes.)
2. Describe the complete video - bibliographic information (title, creator, dates, subjects, item numbers, publisher details, names, synopsis etc.) plus format, framerate, duration etc
3. Describe each sequence - id, start time/frame, end time/frame, brief textual summary
4. Describe each scene - id, start time/frame, end time/frame, brief textual summary, transcript (ideally derived from a closed caption decoder)
5. Describe each shot - id, start time/frame, end time/frame,keyframe (first frame of the shot, ideally derived from an automatic shot detection algorithm)

2.1 Example of Indexing a Typical Museum Video Clip

Below is an example of the indexed breakdown of a 60 second video clip on the wreck of the Pandora, recorded by the Qld Museum [1]. The clip consists of 4 scenes, each of which contains a number of shots. Associated with each scene is an ID, a brief description, its duration (SMPTE time codes) and its associated transcript. Associated with each shot is an ID, a brief description, the start time code and a GIF image which is the first frame (keyframe) from that shot. We also assume that this clip is the third sequence in a fictitious epsiode of Quantum, the ABC's science documentary program.

Sequence #3

Scene#3.1 - The Pandora's Place in History
Duration = 19:31:24;1 - 19:31:35;25 (12secs)
Transcript = "HMS Pandora was the Royal Navy warship sent to the South Pacific to capture the Bounty mutineers. She left England in November 1790 under Captain Edward Edwards."

Shot#3.1.1	**Shot#3.1.2**
A Reproduction of the Pandora	Image of Captain Edwards
19:31:24;1	19:31:30;1

Scene#3.2 - The Shipwreck
Duration = 19:31:36;1 - 19:31:53;25 (18secs)
Transcript= "The wreck is located about 120 km east of Cape York.. It is evident that the hull was intact when it sank but it has gradually been buried by accumulating layers of coralline sand. "

Shot#3.2.1	**Shot#3.2.2**	**Shot#3.2.3**
Protruding anchor	Plan of the Wreck	The excavation process.
19:31:36;1	19:31:42;1	19:31:48;1

Scene#3.3 - Artifacts at the Qld Museum
Duration = 19:31:54;1 - 19:32:11;25 (18secs)
Transcript = "The Pandora wreck has surrendered a wealth of significant artefacts which help to paint a picture of naval life at sea in the late 18th century."

Shot#3.3.1	**Shot#3.3.2**	**Shot#3.3.3**
Numerous bottles and containers.	The Surgeon's gold fob watch.	An officer's bowl.
19:31:54;1	19:32:00;1	19:32:06;1

Scene#3.4 - The Expeditions
Duration = 19:32:12;1 - 19:32:23;25 (12secs)
Transcript = "So far eight archeological expeditions have been carried out. At least three , possibly four more expeditions are planned to be completed by 2001."
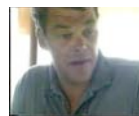
Shot#3.4.1	**Shot#3.4.2**	**Shot#3.4.3**
The Pacific Conquest	Peter Gesner, Expedition Leader	Recovering a Canon
19:32:12;1	19:32:16;1	19:32:20;1

3. 3. Extensions to Dublin Core for Moving Images

The elements of Dublin Core are: Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage and Rights. The semantics of these attributes are described in [2].

The following is the list of sub-elements at the time of writing this paper. This list is still under development by the Dublin Core community.

- Title.Main ,Title.Alternative
- Creator.PersonalName, Creator.PersonalName.Address, Creator.CorporateName, Creator.CorporateName.Address
- Publisher.PersonalName, Publisher.PersonalName.Address, Publisher.CorporateName, Publisher.CorporateName.Address
- OtherContributor.PersonalName, OtherContributor.PersonalName.Address, OtherContributor.CorporateName, OtherContributor.CorporateName.Address
- Date.Created, Date.Issued, Date.Available, Date.Acquired, Date.DataGathered, Date.Accepted, Date.Valid
- Relation.IsPartOf, Relation.HasPart, Relation.IsVersionOf, Relation.HasVersion, Relation.IsFormatOf, Relation.HasFormat, Relation.References, Relation.IsReferencedBy, Relation.IsBasedOn, Relation.IsBasisFor, Relation.Requires, Relation.IsRequiredBy
- Coverage.PeriodName, Coverage.PlaceName, Coverage.T, Coverage.X, Coverage.Y, Coverage.Z, Coverage.Polygon, Coverage.Line, Coverage.3D
  The semantics for these attributes are described in [3].

3.1 Moving Image Resources Workshop Recommendations

The Resource Discovery Workshop: Moving Image Resources [4], examined Dublin Core's potential use for describing moving images resources, tested it against a variety of examples, and critically reviewed its application. It concluded that the Dublin Core model could be used to describe moving image resources given some provisos and solutions to the problems listed below:

- Dublin Core terminology is not sufficiently intuitive for non-library trained researchers and non-specialists to use. To overcome this, ample qualifiers (i.e. long definitive lists of sub-elements and Schemes) should be provided.
- Dublin Core has difficulty satisfying the widely differing needs of both non-specialist interdisciplinary searchers and specialist users.
- *DC.Publisher* requires a large number of sub-elements for moving image resources, including place.
- To overcome the problem of separating primary from secondary creators, *DC.Creator* and *DC.Contributor* should be combined into *DC.Creator* with a large number of clearly specified sub-elements indicating the role.
- Differentiating between original works, various manifestations during production and digital surrogates and each of their respective *DC.Creator*, *DC.Publisher*, *DC.Date* values is a major problem.
- *DC.Coverage* shouldn't be used at all since it can't be used consistently to contain concepts of place and duration. Place can be allocated to either *DC.Subject*

(provenance) or *DC.Publisher* (place of release) and duration (running time) can be allocated to *DC.Format*.
- Only *DC.Description*, the free text description does not potentially require some kind of sub-element or Scheme, apart from the suggestion that censorship board classification should go here.

A summary of the outcomes of this workshop can also be found at [5].

### 3.2 Proposed Dublin Core Extensions for Multilevel Searching

The Moving Image Workshop focused primarily on the semantics of what bibliographic data should be put in which Dublin Core element. Defining what to put where and the lists of sub-elements and Schemes required to satisfy different communities' semantical needs is best left to the specialists themselves.

A major but different type of problem identified by the workshop is the one of satisfying the differing needs of non-specialist interdisciplinary searchers and specialist users. Some users require only very basic information whilst others require detailed interpretive descriptions at a very low level. One of the most problematic issues with trying to apply a "core" data set to something as complex as film or video, is that even summary information can typically include a brief interpretative description of every shot, a full cast and credits list, details of awards and copyright details and detailed technical information often running to many hundreds of lines of data entry. The Moving Image Resources Workshop identified a need for some distinction between 'core' and 'full' data sets for moving image resources. In many cases archival catalogue records are so detailed that they provide a surrogate to actually viewing the resource. This can be particularly important where viewing might endanger a fragile original or for academic researchers who may not need to view a film but do need to find detailed information about it.

The following section describes a solution to this problem through the use of optional extensions to certain Dublin Core elements. This approach provides multiple levels of descriptive information. The top level can be used for non-specialist inter-disciplinary searching. The lower levels can be used for fine-grained discipline-specific searching. The elements discussed are Type, Description, Format, Relation and Coverage.

3.2.1 DC.Type
This defines the category of the resource. For the sake of interoperability, *Type* should be selected from a hierarchy of enumerated lists. For example:
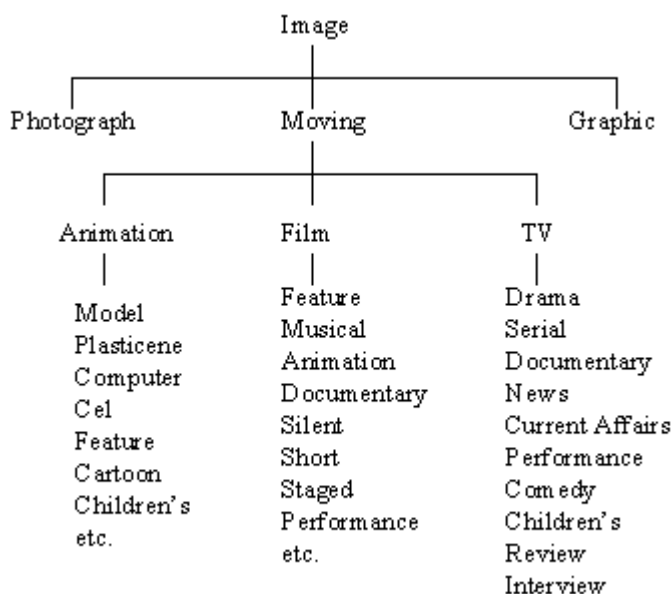
The structured lists above enable the genre of the complete clip/document to be specified. In addition, there is a need to be able to specify parts and sub-parts of complete clips/documents.

Generally film and video documents can be broken down into the following parts: sequences, scenes, shots, frames. Each sequence consists of a number of contiguous scenes. Each scene consists of a number of contiguous shots. Each shot consists of a number of contiguous frames. Each frame can be subdivided into regions representing actors or objects. This hierarchy of enumerated types also defines the rules for valid Relations between Types i.e. *IsPartOf* and *HasPart*.

- Sequence
  - Scene
    - Shot
      - Frame
        - Object/Actor/Person

Some examples of Types based on the enumerated lists above are:
- DC.Type = "Image.Moving.Film.Documentary.sequence.scene"
- DC.Type = "Image.Moving.TV.News.sequence.scene.shot.frame"

3.2.2 DC.Description

Currently within Dublin Core, this represents a textual description of the content of the resource. It is usually an abstract in the case of document-like objects or a textual content description in the case of visual resources. In reality, the description can be any media type e.g. text, image, audio, video, or a URI.

In the video clip indexing example described in , the complete sequence/clip, the scenes and the shots possess a textual description. In addition, scenes possess a transcript and shots possess a keyframe. This paper proposes that each *DC.Type* possess an associated set of allowable descriptors which are specified as subelements to the *DC.Description* element.

For example if DC.Type = "Image.Moving.TV.Documentary.Scene" then valid descriptors are Description.text and Description.transcript. If DC:Type = "Image.Moving.TV.Documentary.Scene.Shot" then valid descriptors are Description.text and Description. keyframe. If DC.Type = "Image.Moving.TV.Documentary.Scene.Shot.Frame" then valid descriptions are Description.text and Description.histogram which is a colour histogram of the frame.

In addition, the valid format of the content of a particular description type must match a value from the IMT (Internet Media Type) Scheme. For example the value of Description.keyframe value must be one of the IMT image formats: image/gif, image/jpeg, etc.

Alternatively, the actual content can be a value taken from an enumerated list or controlled vocabulary specified by a given Scheme. For example, Camera Motion must be selected from one of: *dolly forward, dolly back, truck left, truck right, pan left, pan right, tilt up, tilt down, zoom in, zoom out, stationary*. Camera Distance must be one of: *close-up*, *medium shot* or *long shot*. Camera Angle must be either *low*, *high* or *eye-level*. Opening and Closing transitions can only be one of: *cut*, *fade*, *wipe* or *dissolve*.

Table 1 below illustrates the proposed hierarchical structure and examples of associated permissable descriptors and formats. This approach is suffiently flexible to allow particular communities and working groups to define their own rules on combinations of descriptors and descriptor schemes.

Table 1. Resource Types and Permissable Descriptor Types and Formats

DC.Type	DC.Description	Allowable Formats
Image.Moving.*	DC.Description.Text	Text
Image.Moving.*. sequence	DC.Description.Text	Text
Image.Moving.*. sequence.scene	DC.Description.Text	Text
	DC.Description.Script	Text
	DC.Description.Transcript	Text
	DC.Description.EditList	Text
	DC.Description.Duration	secs, frames
	DC.Description.StartTime	secs, frame no, SMPTE
	DC.Description.EndTime	secs, frame no, SMPTE
	DC.Description.Keyframe	JPEG, GIF
	DC.Description.Locale	Text
	DC.Description.Cast	Text
	DC.Description.Objects	Text
Image.Moving.*. sequence.scene.shot	DC.Description.Text	Text
	DC.Description.Duration	secs, frames
	DC.Description.StartTime	secs, frame no, SMPTE
	DC.Description.EndTime	secs, frame no, SMPTE
	DC.Description.Keyframe	JPEG, GIF
	DC.Description.Camera.Dist	Controlled vocab.
	DC.Description.Camera.Angle	Controlled vocab.
	DC.Description.Camera.Motn	Controlled vocab., line
	DC.Description.Lighting	Controlled vocab.
	DC.Description.OpenTrans	Controlled vocab.
	DC.Description.CloseTrans	Controlled vocab.
Image.Moving.*. sequence.scene.shot. frame	DC.Description.Text	Text
	DC.Description.Image	JPEG,GIF
	DC.Description.Timestamp	secs, frame no, SMPTE
	DC.Description.Colour	Histogram, Text
	DC.Description.Anno.Text	Text
	DC.Description.Anno.Posn	Point, Area, Object-Id
Image.Moving.*. sequence.scene.shot. frame.object	DC.Description.Text	Text

	DC.Description.Position	Point
	DC.Description.Shape	Polygon
	DC.Description.Trajectory	Line
	DC.Description.Speed	Pixels/frame
	DC.Description.Colour	Histogram, Text
	DC.Description.Texture	Tamura,SAR feature vector
	DC.Description.Volume	3D polygon
	DC.Description.Anno.Text	Text
	DC.Description.Anno.Posn	Point, Area

3.2.3 DC.Format

This represents the data format of the resource and can be used to identify the software and possibly hardware that might be needed to display or operate the resource. For the sake of interoperability, Format should be selected from an enumerated list that is currently under development in the Dublin Core workshop series. The kinds of information which will be stored in this element include:

> Format.video.type = 35mm film, VHS etc.
> Format.video.colourdepth = 256
> Format.video.length = 31 mins.
> Format.video.codec = MJPEG, MPEG1, MPEG2, AVI, QT, etc.
> Format.video.framerate = 25
> Format.video.resolution, Format.video.width, Format.video.height
> Format.sound, Format.sound.channels, Format.sound.samplerate

3.2.4 DC.Relation

For video, we need to be able to describe parts of complete videos or clips such as: *sequences*, *scenes* and *shots*. The Relation subelements *HasPart* and *IsPartOf* provide this facility. For example the Relation values for scene3.3 would be:
Relation.HasPart Content= shot3.3.1, shot3.3.2, shot3.3.3
Relation.IsPartOf Content= sequence3
The hierarchy of parts and sub-parts will impose rules on the use of the *HasPart* and *IsPartOf* subelements. Clearly shots can be parts of scenes but not vice versa.

3.2.5 DC.Coverage

For moving image data, the proposal is to use the Coverage element to describe the temporal location of clips, scenes, shots etc. within a larger video segment. The format of the time value may be a frame number, SMPTE time code or time from the start.
Coverage.t.min scheme=SMPTE content="09:45:23;14"
Coverage.t.max scheme=SMPTE content="09:45:32;1"
In addition, the Coverage subelements, Coverage.x, Coverage.y, Coverage.z, Coverage.line, Coverage.polygon and Coverage.3D can be used to describe the spatial

locations, motion and shapes of objects/actors within a frame. Detailed descriptions of these subelements, as determined by the Coverage Working Group can be found at [6].

4. Application of Dublin Core Extensions to Museum Clip Indexing

The following section provides an example of how Dublin Core, with the extensions described above, could be applied to index the museum clip described in Section 2.1.

We assume that the clip chosen is the third sequence in a fictitious episode of the 30 minute documentary program, Quantum. This particular sequence contains 4 scenes, each of which contains a number of shots. Only the Dublin Core elements for Scene 3.3 and Shot 3.3.2 are described. The descriptions for the other scenes and shots can easily be deduced from this example.

Complete Documentary Program
    Title = "Quantum"
    Creator = "Australian Broadcasting Service"
    Publisher = "Australian Broadcasting Service"
    Contributor.Presenter = "Adam Spencer"
    Description.text = "A weekly half hour science program"
    Date = 1998-02-20
    Type = "Image.Moving.TV.documentary"
    Format.type = VHS
    Format.length = 30 mins
    Identifier = "http://www.abc.com.au./quantum/98-02-20.mpg"
    Language = en

Sequence#3
    Subject = "Pandora (Frigate); Shipwrecks -- Queensland; Underwater archaeology"
    Description.text = "An overview of the wreck of HMS Pandora and the excavation expeditions being carried out by the Queensland Museum"
    Contributor.Reporter = "Adam Spencer"
    Type = "Image.Moving.TV.documentary.sequence"
    Format.length = 60 secs
    Coverage.t.min scheme=SMPTE content= 19:31:24;1
    Coverage.t.max scheme=SMPTE content= 19:32:24;1
    Relation.IsPartOf = Complete Documentary Program
    Relation.HasPart = scene3.1, scene3.2, scene3.3, scene3.4

Scene#3.3
    Description.text ="Artifacts at the Queensland Museum"
    Description.transcript = "The Pandora wreck has surrendered a wealth of significant artefacts which help to paint a picture of naval life in the late 18th

century."
Type = "Image.Moving.TV.documentary.sequence.scene"
Format.length = 18 secs
Coverage.t.min scheme=SMPTE content= 19:31:54;1
Coverage.t.max scheme=SMPTE content= 19:32:11;25
Relation.IsPartOf = sequence3
Relation.HasPart = shot3.3.1, shot3.3.2, shot3.3.3

Shot#3.3.2
Description.keyframe = shot3.3.2.gif
Description.text = "The surgeon's gold fob watch."
Type = "Image.Moving.TV.documentary.sequence.scene.shot"
Format.length = 6 secs
Coverage.t.min scheme = SMPTE content = 19:32:00;1
Coverage.t.max scheme = SMPTE content = 19:32:05;25
Relation.IsPartOf = scene3.3

4.1 The Resource Description Framework

The Resource Description Framework (RDF) [7] is a specification currently under development within the W3C Metadata activity [8]. RDF is designed to provide an infrastructure to support metadata across many web-based activities. It is the result of a number of metadata communities bringing together their needs to provide a robust and flexible architecture for supporting metadata on the Internet and WWW. It's design has been heavily influenced by the Warwick Framework work [9].

RDF will allow different application communities to define the metadata property set that best serves the needs of each community. It will provide a uniform and interoperable means to exchange the metadata between programs and across the Web. RDF will also provide a means for publishing both a human-readable and a machine-understandable definition of the property set itself.

RDF is still under development but to date the following documents have been released for public comment:
- A public draft of the RDF Model and Syntax Specification (released Feb. 16 1998) [10].
- A public draft of the RDF Schema work-in-progress (released April 10 1998) [11].

RDF uses XML (eXtensible Markup Language) [12], as the transfer syntax in order to leverage other tools and code bases being built around XML. For example, SMIL (Synchronized Multimedia Integration Language) [13], a language for specifying Web-based Multimedia presentations, is encoded in XML.

We have chosen to use the RDF syntax for encoding video metadata because it provides a model for defining relationships between resources. This is illustrated below. The layered video structure is supported by defining RDF Sequence collection nodes within each DC:Relation:HasPart and a separate RDF:Description for each

element of the Sequence collection. The indentations contribute to the readability and ease of understanding of the video structure.

More examples of the use of RDF syntax to encode Dublin Core metadata can be found at [16].

4.2 Dublin Core Example in RDF

Below are a series of RDF-encoded metadata descriptions for the different levels of the video clip. Each RDF description points to the corresponding actual content via the *About* value, which is a URL.

The difficulty with continuous media is that there is currently no standard way of pointing to a particular portion of an audio or video file, using a URL. Qualifying information that needs to be able to be specified in a URL referring to video or audio content includes:

- a specific time offset into a video/audio
- a specific time range within a video/audio
- a specific label within a video/audio where the label is resolved to a position and duration within the video/audio by some other service

SMIL allows you to define a link to a fragment of a video source by defining a anchor element with specific begin and end attributes e.g.

```
<video src="http://www.abc.com.au/quantum/98-02-20.mpg">
  <anchor id="seq3" begin="00:54:24.01" end="00:56:32.25"/>
</video>
```

Using this approach, we can refer to sequence#3 by:
  "http://www.abc.com.au/quantum/98-02-20.mpg#seq3"

The RDF metadata for the URL "http://www.abc.com.au/quantum/98-02-20.mpg" is shown below.

```
<?xml:namespace ns="http://www.w3c.org/RDF/" prefix="RDF"?>
<?xml:namespace ns="http://metadata.net/DC/" prefix="DC"?>

<RDF:RDF>
  <RDF:Description About="http://abc.com/98-02-20.mpg">
    <DC:Title>Quantum</DC:Title>
    <DC:Creator>Australian Broadcasting Service</DC:Creator>
    <DC:Subject>Science, Documentary</DC:Subject>
    <DC:Description>A weekly half hour science
  program</DC:Description>
    <DC:Publisher>Australian Broadcasting Service</DC:Publisher>
    <DC:Contributor.Presenter>Adam Spencer
                              </DC:Contributor.Presenter>
    <DC:Format DC:Scheme="IMT">video/mpg</DC:Format>
    <DC:Type>Image.Moving.TV.Documentary</DC:Type>
    <DC:Language>en</DC:Language>
    <DC:Date>1998-02-20</DC:Date>
    <DC:Format.Length>30 mins</DC:Format.Length>
    <DC:Relation.HasPart>
      <RDF:Seq>
        <RDF:LI Resource="http://abc.com/98-02-20.mpg#seq1"/>
        <RDF:LI Resource="http://abc.com/98-02-20.mpg#seq2"/>
        <RDF:LI Resource="http://abc.com/98-02-20.mpg#seq3"/>
        <RDF:LI Resource="http://abc.com/98-02-20.mpg#seq4"/>
        <RDF:LI Resource="http://abc.com/98-02-20.mpg#seq5"/>
        <RDF:LI Resource="http://abc.com/98-02-20.mpg#seq6"/>
        <RDF:LI Resource="http://abc.com/98-02-20.mpg#seq7"/>
      </RDF:Seq>
```

The RDF metadata for the URL "http://www.abc.com.au/quantum/98-02-20.mpg#seq3" is shown below. Similarly, the metadata for the scenes and shots can be deduced from these examples.

```
<?xml:namespace ns="http://www.w3c.org/RDF/"prefix="RDF"?>
<?xml:namespace ns="http://metadata.net/DC/"prefix="DC"?>
<RDF:RDF>
  <RDF:Description About= "http://abc.com/quantum/98-02-
20.mpg#seq3">
     <DC:Type>Image.Moving.TV.documentary.sequence</DC:Type>
     <DC:Description.text>An overview of the wreck of HMS Pandora
  and the excavation expeditions being carried out by the
  Queensland Museum.</DC:Description.text>
    <DC:Subject>Pandora(Frigate);Shipwrecks -- Queensland;
 Underwater archaeology </DC:Subject>
     <DC:Contributor.Reporter>Adam Spencer
                   </DC:Contributor.Reporter>
     <DC:Format.Length>60 secs</DC:Format.Length>
     <DC:Coverage.t.min DC:Scheme="SMPTE">19:31:24;1
 </DC:Coverage.t.min>
     <DC:Coverage.t.max DC:Scheme="SMPTE">19:32:24;1
 </DC:Coverage.t.max>
    <DC:Relation.HasPart>
      <RDF:Seq>
        <RDF:LI Resource="http://abc.com/quantum/98-02-
   20.mpg#scene3.1"/>
        <RDF:LI Resource="http://abc.com/quantum/98-02-
   20.mpg#scene3.2"/>
        <RDF:LI Resource="http://abc.com/quantum/98-02-
   20.mpg#scene3.3"/>
        <RDF:LI Resource="http://abc.com/quantum/98-02-
   20.mpg#scene3.4"/>
      </RDF:Seq>
    </DC:Relation.HasPart>
   </RDF:Description>
</RDF:RDF>
```

5. Audio Metadata

So far, only visual and textual indexing have been considered, but audio also
constitutes a major source of indexing information. Speech recognition can enable
keyword queries on videos without the need for transcripts. By providing an example
of a particular speaker's speech, speaker recognition enables users to perform queries
such as: "Find all video clips of Janette Howard speaking". Music recognition can
enable the retrieval of videos containing a particular tune by humming or whistling.
Audio cues such as silence, music and volume can be used to assist with the video
segmentation. The downside of including this audio information is that it adds even
further complexity to the already complex video metadata.

Figure 1 below illustrates how the soundtrack adds more layers to the already hierarchical video structure. Now the video consists of both temporally parallel and sequential components.
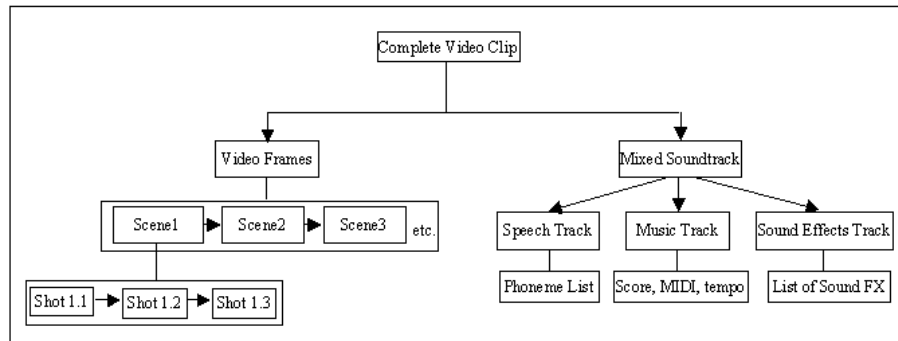

Fig. 1. Multilayered Hierarchical Structure of a Video Clip

The sound track plays back in parallel with the playback of the video frames. The soundtrack may consist of a large number of individual sound tracks mixed together. Typical types of soundtracks include : speech, music, sound effects, live, mixed. If the individual speech, music and sound effects tracks are not available, then mixed sound tracks can potentially be seperated into speech, music and sound effects tracks. Each of these individual sound tracks can be described using their own domain-specific descriptors and descriptor schemes and if required can be segmented into scenes and shots. For example the speech track may be described by a list of phonemes and their durations or phone lattices. A music track may be described by a score, MIDI file, melodic contour, frequency contour, tempo or amplitude envelope. A sound effects track may be described by a list of sound effect objects.

Because audio is such a complex data structure in its own right, this paper will not attempt to describe the possible DC:Description subelements and formats corresponding to each of the five types: mixed, speech, music, soundeffects, live. However it will briefly discuss the various approaches for including the audio metadata within the complete video metadata to enable cross-modal searching.

5.1 Adding Audio Metadata Using RDF

So far all of the video structures have been temporally sequential. The inclusion of audio metadata adds the requirement for temporal parallelism. RDF only provides three types of collections :*sequence*, *bag* and *alternatives*. *Sequence* can be used to specify ordering between collection members e.g. temporal, importance, alphabetical. *Bag* implies that all of the members are of equal importance. *Alternatives* implies there is a choice between members. Since there is no specific *Parallel* element for describing such temporal relationships, the next best alternative is to use the Bag element and to specify temporal locations and durations using DC.Coverage.t.min and

DC.Coverage.t.max or via a SMIL temporal fragment anchor. The ability to specify synchronisation between specific components is not supported without further subelements.

Below is a simple example which illustrates how to specify the temporal relationships and metadata of both audio and video components using RDF. The descriptions can be specified using Dublin Core, as shown above, or any other domain-specific metadata format.

```
<?xml:namespace ns="http://www.w3c.org/RDF/" prefix="RDF"?>
<?xml:namespace ns="http://metadata.net/DC/" prefix="DC"?>

<RDF:RDF>
    <RDF:Description About="http://abc.com/quantum/98-02-
20.mpg">
        <DC:Title>Quantum</DC:Title>
        <DC:Creator>ABC</DC:Creator>
        <DC:Subject>Science Documentary</DC:Subject>
        <DC:Description>A weekly half hour science program.
        </DC:Description>
        <DC:Publisher>ABC</DC:Publisher>
        <DC:Format DC:Scheme="IMT">video/mpg</DC:Format>
        <DC:Type>Image.Moving.TV.Documentary</DC:Type>
        <DC:Language>en</DC:Language>
        <DC:Date>12/05/98</DC:Date>
        <DC:Format.Length>30 mins</DC:Format.Length>
        <DC:Relation.HasPart>
            <RDF:Bag BAGID="CompleteVideo">
             <RDF:LI Resource= "http://abc.com/quantum/98-02-
20.mpg#sndtrack1"/>
                <RDF:LI Resource= "http://abc.com/quantum/98-02-
20.mpg#sndtrack2"/>
                <RDF:LI Resource= "http://abc.com/quantum/98-02-
20.mpg#sndtrack3"/>
                <RDF:LI Resource= "http://abc.com/quantum/98-02-
20.mpg#sndtrack4"/>
                <RDF:LI Resource= "http://abc.com/quantum/98-02-
20.mpg#videopart"/>
            </RDF:Bag>
        </DC:Relation.HasPart>
    </RDF:Description>
```

The metadata for the URL "http://www.abc.com.au/quantum/98-02-20.mpg#videopart" is as for the example in Section 4.2.

6. The Pros and Cons of Using Dublin Core and RDF for Video Metadata

The advantages of a pure Dublin Core approach include:
- It provides both 'core' and 'full' data descriptions to satisfy a range of user groups' needs.
- It enables searching across different media types and can exploit all of the work already done on Dublin Core metadata generation and Dublin-Core based indexing and search tools.
- It inherits the advantages associated with Dublin Core - simplicity, semantic interoperability, scalability, international consensus and flexibility. (Though it could justifiably be argued that the proposed extensions for video destroy the simplicity.)

The advantages associated with using RDF syntax for encoding the Dublin Core metadata are:
- It allows labelled directed graphs to be built which support the hierarchical containment structure of video.
- It is encoded in XML (eXtensible Markup Language) which is based on SGML and is better able to support multimedia than HTML.
- It can leverage off other tools and code bases being built around XML e.g. SMIL (Synchronized Multimedia Integration Language) [13], a declarative language for describing Web-based multimedia presentations. (SMIL describes how the various components are to be combined temporally and spatially to create a presentation.)
- It is both human-readable and machine-readable.
- It provides a container for different communities' metadata schemes.

Dublin Core was designed to do high-level interdisciplinary searching for complete textual documents across heterogeneous databases and schemas. It provides a simplified set of 15 elements which enables searching across the WWW. Dublin Core was not designed to provide metadata at a low level such as scenes and shots. Consequently there are a number of disadvantages associated with using Dublin Core for describing complex video documents. These include:
- The loss of simplicity.
- The need for a great number of sub-elements (especially within the Description element), Schemes and rules.
- There is no way to specify fine-grained synchronization between the different components i.e. explicit durations or absolute and relative offsets.
- The entanglement of semantics and structure between Dublin Core and RDF. There is no clear delineation between semantics in Dublin Core and video structure in RDF.

The last issue of separation of structure from semantics is problematic. For the sake of simplicity, it would be better if the two components could be separated. But the

relation-ships between elements is often an important part of the metadata and thus the structural descriptions need to be integrated with the semantic descriptions as part of the metadata. This can lead to messy, complex metadata that is not easily read.

The above exercise also revealed a number of limitations associated with using RDF to contain Dublin Core video metadata descriptions. These include:

- It is unclear whether RDF permits nested collections i.e. collections within collections, as illustrated in the RDF code below.
- It is unclear how or if RDF allows pointers to metadata (i.e. another rdf file) rather than a resource (e.g. an mpg file)
- RDF does not provide *Par* or Parallel-type Collections in which each of the components are replayed in parallel.
- RDF doesn't support the specification of fine-grained synchronization between elements i.e. explicit durations or temporal offsets.

7. SMIL

SMIL (Synchronized Multimedia Integration Language) [13] is a declarative language for describing Web-based multimedia presentations. SMIL describes how the various components are to be combined temporally and spatially to create a presentation. Its objectives are very similar to the relatively complex HyTime [14] and MHEG [15] standards, which are based on SGML, but because SMIL is based on XML, it is much simpler to use. Although SMIL was designed to describe combinations of multimedia components, it could also be used to deconstruct a composite multimedia document (such as a video clip with sound) and to describe the temporal and spatial structure of its components.

SMIL describes four fundamental aspects of a multimedia presentation:

- temporal specifications: primitives to encode the temporal structure of the application and the refinement of the (relative) start and end times of events;
- spatial specifications: primitives provided to support simple document layout;
- alternative behavior specification: primitives to express the various optional encodings within a document based on systems' or user requirements; and
- hypermedia support: mechanisms for linking parts of a presentation.

This paper is primarily concerned with the temporal specifications of SMIL. SMIL provides coarse-grain and fine-grain declarative temporal structuring of an application. Coarse grain temporal information is given in terms of two structuring elements:

- <seq> ... </seq>: A set of objects that occur in sequence.
- <par> ... </par>: A collection of objects that occur in parallel.

Elements defined within a <seq> group have the semantics that a successor is guaranteed to start after the completion of a predecessor element. Elements within a <par> group have the semantics that, by default, they all start at the same time. Once started, all elements are active for the time determined by their encoding or for an explicitly defined duration. Elements within a <par> group can also be defined to end at the same time, either based on the length of the longest or shortest component or on the end time of an explicit master element. Note that if objects within a <par> group are of unequal length, they will either start or end at different times, depending on the attributes of the group.

Fine grain synchronization control is specified in each of the object references through a number of timing control relationships:

- explicit durations: a DUR=" length " attribute can be used to state the presentation time of the object;
- absolute offsets: the start time of an object can be given as an absolute offset from the start time of the enclosing structural element by using a BEGIN=" time " attribute;
- relative offsets : the start time of an object can be given in terms of the start time of another sibling object using a BEGIN=" object_id + time " attribute.

At present, only explicit time offsets into objects are supported, but a natural extension is to allow content markers, which provide content-based tags into a media object.

7.1 Example SMIL Code

Below is an example of a SMIL description of the museum clip in Section 1.1. Sequence#3 consists of 4 sequential scenes. Each scene consists of video and audio playing in parallel.

```
<smil sync="soft">
   <head>
    <layout type="text/smil-basic">
      <channel id="v-main" left="5%" top="5%" width="90%"
height="90%"/>
      <channel id="audio"/>
      <channel id="music"/>
    </layout>
   </head>
   <body>
     <seq id="Sequence#3">
      <par id="scene1">
        <video id="intro" channel="v-main"   src="mpeg/history.mpg"/>
        <audio id="intro_voiceover" channel="audio"
src="audio/abc/intro.aiff" begin="1.5s"/>
        <audio id="leader_music" channel="music"
src="audio/logo1.aiff"/>
      </par>
      <par id="scene2">
       <video id="shipwreck" channel="v-main"
src="mpeg/shipwreck.mpg"/>
      </par>
      <par id="scene3">
       <video id="artifacts" channel="video" src="mpeg/artifacts.mpg"
dur="16.0s"/>
        <audio id="voiceover" channel="audio"
src="audio/abc/artifacts.aiff" begin="0.9s"/>
      </par>
      <par id="scene4">
       <video id="expeditions" channel="v-main"
src="mpeg/expeditions.mpg"/>
        <audio id="trailer_voiceover" channel="audio"
src="audio/abc/wrapup.aiff"/>
        <audio id="trailer" channel="music" src="audio/logo2.aiff"
begin="id(expeditions)(begin)+3.5s"/>
      </par>
    </seq>
   </body>
  </smil>
```

7.2 Applying SMIL to Multimedia Metadata

There are two ways in which SMIL can be applied to video metadata. They are:

1. Adding metadata via the SMIL "meta" attribute.

In this case, the multimedia content points to the metadata. Every SMIL element has an optional meta attribute. Ideally this could be a pointer to an RDF file. For example:

```
<video id="wrapup" channel="v-main" src="mpeg/artifacts.mpg"
meta="http://www.Qmuseum/videodesc/artifacts.rdf">
```

2.Extending RDF by adding synchronisation and timing controls.

By adding a Par (parallel) Collection Type to RDF and using the SMIL DUR, BEGIN and END attributes then both coarse and fine-grained temporal structuring would be possible within RDF. For example:

```
<?xml:namespace href="http://www.w3c.org/RDF/" as="RDF"?>
<?xml:namespace href="http://purl.org/RDF/DC/" as="DC"?>
<?xml:namespace href="http://www.w3c.org/TR/WD-smil/" as="SMIL"?>
<RDF:RDF>
    <RDF:Description       RDF:HREF="http:/abc.com/quantum/98-02-
20.mpg">
               .
               .
      <DC:Relation.HasPart>
        <RDF:Par BAGID="CompleteVideo">
          <RDF:LI ID="VideoPart"
            RDF:HREF="http://abc.com/quantum/pandora.mpg"/>
          <RDF:LI ID="SoundTrack1"
            RDF:HREF="http://abc.com/music/opening.wav"
   SMIL:DUR="6.0s" SMIL:BEGIN="ID(VideoPart)(BEGIN)+1.8s"/>
           <RDF:LI ID="SoundTrack2"
            RDF:HREF="http://abc.com/audio/intro.aiff"
   SMIL:DUR="4.0s" SMIL:BEGIN="ID(VideoPart)(BEGIN)+2.8s"/>
        </RDF:Par>
      </DC:Relation.HasPart>
   </RDF:Description>
</RDF:RDF>
```

8. Current State of MPEG-7

The objective of MPEG7 [17] is to provide standardized descriptions of audiovisual information to enable it to be quickly and efficiently searched. MPEG-7, formally called `Multimedia Content Description Interface', will standardize:
- A set of description schemes and descriptors, and
- A language to specify description schemes, i.e. a Description Definition Language (DDL)

MPEG-7 will address the coding of these descriptors and description schemes. The combination of descriptors and description schemes shall be associated with the content itself, to allow fast and efficient searching for material of a user's interest. AV material that has MPEG-7 data associated with it, can be indexed and searched for. This `material' may include: still pictures, graphics, 3D models, audio, speech, video, and information about how these elements are combined in a multimedia presentation (`scenarios', composition information).

The development of the MPEG-7 standard is still at a very early stage with the Call for Proposals being scheduled for October 1998 and the Draft International Standard not expected to be published until July 2001. But given the overlap in objectives between MPEG-7 and Dublin Core, it makes sense for the MPEG-7 community to be aware of the work of the Dublin Core community and vice versa, to ensure compatibility, interoperability and mappability where possible.

8.1 Hybrid Approach

Minimalists from the Dublin Core community will undoubtedly be offended by the proposal to extend Dublin Core to such fine-grained descriptions as outlined above. A hybrid proposal based on RDF would overcome such criticisms but still exploit the valuable aspects of Dublin Core. RDF was designed to provide a container for different metadata formats. The proposal is to use RDF to contain both Dublin Core and MPEG7 descriptions of the same content.

Dublin Core can be used to describe audiovisual documents as a whole and to enable searching for complete audiovisual documents i.e. search and query at a high level on the 15 core elements. For example; "Find all video clips on Boris Yeltsin". This would perform a text search on the 15 core elements for the string "Boris Yeltsin".

MPEG7 can be used to provide a detailed hierarchical description of the content. The MPEG7 data can be used to enable low level content-based querying such as; "Give me close-up shots of Boris Yeltsin walking in front of the Kremlin". Since large components of the Dublin Core work do satisfy the MPEG7 requirements, it makes sense to exploit these aspects in MPEG-7. The exercise above has shown that

Dublin Core, with extensions (particularly domain-specific qualifiers in the Description field), could form a basis for MPEG-7.

The advantages of the hybrid approach are:

- Existing Dublin Core text-based search engines can still be used to search across heterogeneous media types.
- It satisfies the original intention of Dublin Core to provide a core description and not to replace specialized cataloguing methods.
- Existing catalogues such as US MARC can be mapped to Dublin Core.
- MPEG-7 can be developed independently to provide low level fine-grained content-based querying.
- The easy integration of other developing metadata standards such as PICS (Platform for Internet Content Selection) [18] for classifying audiovisual content.
- SMIL (Synchronized Multimedia Integration Language) can also be used for combining separate audiovisual documents into a synchronized multimedia presentation.

```
<?xml:namespace ns="http://www.w3c.org/RDF/" prefix="RDF"?>
<?xml:namespace ns="http://metadata.net/DC/" prefix="DC"?>
<?xml:namespace ns=http://mpeg.org/mpeg7 prefix="MPEG7"?>
<RDF:RDF>
  <RDF:Description About=  "http://abc.com/98-02-20.mpg">
    <DC:Title>Quantum</DC:Title>
    <DC:Creator>ABC</DC:Creator>
    <DC:Subject>Documentary, Science</DC:Subject>
    <DC:Publisher>ABC</DC:Publisher>
    <DC:Contributor.Presenter>Adam Spencer
                      </DC:Contributor.Presenter>
    <DC:Format DC:Scheme="IMT">video/mpg</DC:Format>
    <DC:Type>Image.Moving.TV.Documentary</DC:Type>
    <DC:Language>en</DC:Language>
    <DC:Date>1998-05-01</DC:Date>
    <DC:Format.Length>30 mins</DC:Format.Length>
    <MPEG7:Duration>1400</MPEG7:Duration>
    <MPEG7:Script>http://abc.com/quantum/98-02-20.txt
        </MPEG7:Script>
    <MPEG7:Locale>Gore Hill Studios</MPEG7:Locale>
  </RDF:Description>
</RDF:RDF>
```

9. Future Work

Future Work includes:
- Extending Reggie, the DSTC Metadata Editor, [19] to generate video metadata. This entails enabling the entry of metadata for multilayered, hierarchical structures. It also requires the definition of a new schema file and the validation of combinations of DC.Type, DC.Description types and DC.Description content.
- Integrating the scene change detection software, closed caption decoder and video replayer and annotator into Reggie.
- Submitting a proposal based on Dublin Core, SMIL, RDF and XML to the MPEG-7 standards committee.
- Building a WWW video search engine based on the metadata repository generated by Reggie. This will involve research into query languages for multimedia metadata.
- Building mappings between high level semantic queries and low level features stored within the video metadata, for specific domains or communities.

Figure 2 illustrates the proposed system setup. Most of the components are available but their integration and enhancements to satisfy certain video requirements are still being carried out.

Output from a TV or VCR is fed into a Closed Caption decoder to generate the transcript. Video output is also fed into an MPEG1 video capture card. The MPEG1 file is input into automatic scene change detection software to generate JPEG images which represent the key frames which occur at the scene changes.

Extensions will be made to the existing DSTC Metadata Editor, Reggie, to enable the generation of standardized metadata descriptions, in RDF format, for each MPEG1 video clip. Reggie will provide the user interface for the user to specify the hierarchical video structure, metadata values and dynamic links and to store all of this in a single standardized RDF machine- and human-readable format.

The generated RDF files are stored in a metadata repository on the HTTP server and the MPEG1 files are stored on the continuous media server. Video delivery is performed via the DSTC's SuperNOVA architecture [20] which provides end-to-end QoS management and streaming video which adapts dynamically to the available bandwidth.

Fig. 2. System Architecture
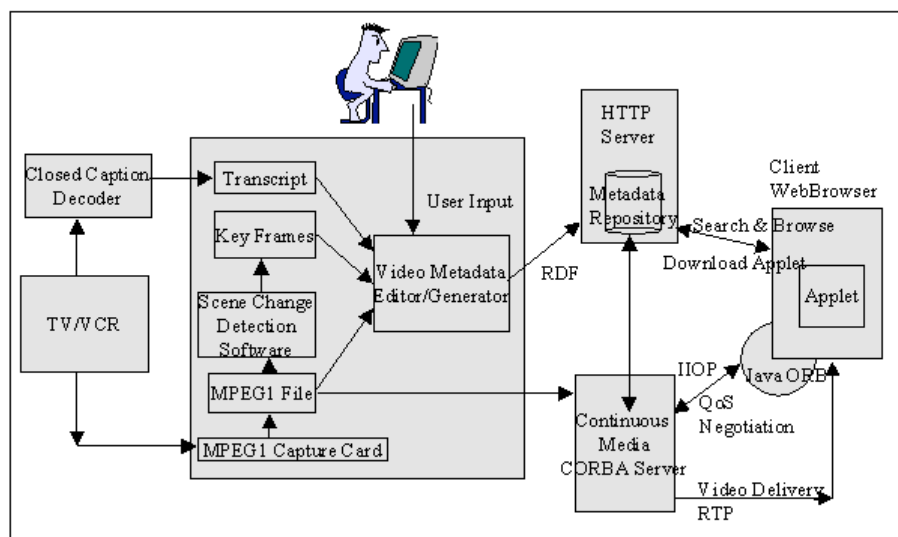
10. Conclusions

With the addition of certain video-specific sub-elements, Dublin Core metadata encoded in RDF (with the addition of timing controls), will satisfy the requirements for indexing most moving image resource types. However this is not what Dublin Core was designed for. It was designed to provide a very simple core of 15 descriptive elements. Minimalists in the Dublin Core community would be horrified at the thought of using Dublin Core extensions to describe something as detailed as an object's texture in a particular frame of a movie.

However, the exercise above shows that the Dublin Core extensions proposed, could provide a good basis for MPEG7. In addition, RDF (with some extensions) provides an ideal infrastructure for describing video using a combination of Dublin Core (for the high level description), MPEG7 (for the lower level fine-grained descriptions) and SMIL for the spatial and temporal structures. The advantages of this approach are many: the output is both machine and human readable; multilayered and hierarchical structures are supported and most compellingly, the work already done on Dublin Core, RDF, SMIL and XML based metadata tools can be exploited.

The development of content-based metadata standards for audiovisual data will provide the key to finding specific content within the rapidly growing complex multimedia archives distributed across museums and other cultural institutions.

Acknowledgements

References

1. Queensland Museum Explorer, 'Dive the Pandora', 1998,
   http://www.Qmuseum.qld.gov.au/culture/pandorawelcome.html
2. 'Description of Dublin Core Elements', http://purl.oclc.org/metadata/dublin_core_elements.
3. Guenther R., 1997, 'Dublin Core Qualifiers/Substructure',
   http://www.loc.gov/marc/dcqualif.html.

4. Duffy C. and Owen C., 1997a., 'Resource Discovery Workshops: Moving Image Resources'
http://pads.ahds.ac.uk/padsUserNeedsMetadataWorkshopsFilm.html.

5. 'A Practical Implementation of the Dublin Core',
http://ahds.ac.uk/public/metadata/disc_11.html.

6. Ad Hoc Working Group – Coverage Element, 1997, 'Dublin Core Element: Coverage',
http://www.sdc.ucsb.edu/~mary/coverage.htm.

7. 'W3C RDF Activity', http://www.w3.org/RDF.

8. 'W3C Metadata Activity', http://www.w3.org/Metadata.

9. Lagoze, C., C.A. Lynch and R. Daniel, 1996, "The Warwick Framework: A Container Architecture for Aggregating Sets of Metadata" ,
http://cs- tr.cs.cornell.edu:80/Dienst/UI/2.0/Describe/ncstrl.cornell/TR96-1593

10. 'W3C Working Draft of the RDF Model and Syntax Specification', 1998,
http://www.w3.org/TR/WD-rdf-syntax.

11. 'W3C Working Draft of the RDF Schema work-in-progress', 1998,
http://www.w3.org/TR/WD-rdf-schema.

12. 'EXtensible Markup Language (XML)', 1998,   http://www.w3.org/XML/.

13. 'Synchronized Multimedia Integration Language', W3C Working Draft, 1998,
http://www.w3.org/TR/WD-smil.

14. 'Papers on HyTime', 1997, http://www.hytime.org/papers/.

15. MHEG-5 Users Group, 1997, http://www.fokus.gmd.de/ovma/mug/.

16. E. Miller and R. Ianella, 'Dublin Core Examples in RDF', 1998,
http://www.dstc.edu.au/RDU/RDF/dc-in-rdf-ex.html.

17. 'MPEG-7 Starting Points and FAQs', 1998, http://www.mpeg.org/~tristan/MPEG/starting-points.html#mpeg7.

18. 'PICS (Platform for Internet Content Selection)', http://www.w3.org/PICS/.

19. 'Reggie, the DSTC Metadata Editor', http://metadata.net/dstc/.

20. 'The SuperNOVA Project', http://www.dstc.edu.au/SuperNOVA/.