

# An XML/RDF Multimedia Specification

Bryan Chafy  
Professor Robert Futrelle  
3/12/1999

## Proposal

A standard (plug-and-play-like) content description and presentation index for multimedia on various media types is needed.

## Justification

As new multimedia formats arise, there is an increasing need for a consistent way to organize them. From the earliest days of computing, folks have put pictures, audio, and text files on standard floppy disks. The most universal way to index such disks was to simply write their contents on the label. This was fine but it did not lend itself to interactivity. Floppy discs generally contained a rather small amount of content as well.

As content becomes more vast, for example network drives, CD's, and DVD's, writing content on a label becomes less practical. A more recent example of this is the MP3 CD. Such CD's can contain 100's of files of content but with only a limited number of methods to index their content. m3u and playlists are examples of such methods. However, they do not apply to mixed content CD's (for example a CD with mp3 files, wav files, and jpeg pictures). In some cases, someone may put a text file at the root level directory of the media, giving it a label like "contents" or "index". This works only for the user and does not adhere to any standard format. An application cannot consistently parse these text files.

## Historical, Other Attempts

### **autorun.inf**

Microsoft has attempted to solve the problem with a file called autorun.inf. When you insert a CD containing this file, the Windows OS will automatically open the file and execute whatever runnable files are contained therein. The problems with such a method are:

1. Only works with a Microsoft OS
2. Currently applies to CD's only
3. autorun.inf is not an indexing format, but rather an auto launching format.

### **CDI**

There have been indexing standards in from years ago that never really took off. The most memorable example would be CDI, invented by Philips Electronics. The primary downfall of CDI was the ridiculous NDA and licensing fees that a content developer had to subject himself

to. The technology was good for its time but it also suffers from being a CD-only format.

CDI lives today only because of the huge Video CD market in China. Note that most computer based VCD players just ignore the CDI and directly play the mpeg video files.

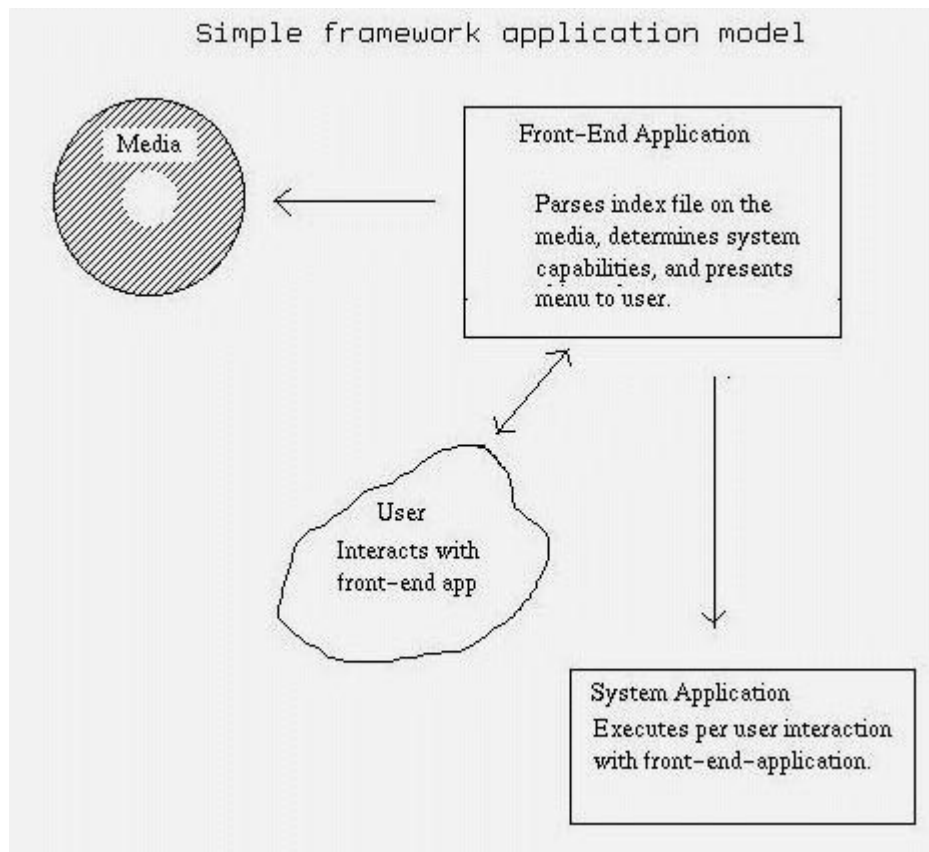
## ID3

ID3 is an attempt at categorizing MP3 files with tagged descriptions [MNID3]. This is useful for MP3 files only. Also, ID3 tags live within the MP3 file itself. This makes searching-for or presenting this information as a list unsuitable.

## Concept

A single file on a media (eg a CD-ROM) shall contain the references to the multimedia files. It represents the presentation and description of those files. This is sometimes called playlist, content, index, catalog, toc, or even readme in some contexts. It is the file one generally looks at to list the contents and get descriptions. This file is usually in the root directory of the media.

The following illustration shows how a front-end application might handle the standard index.



## **Clarification**

This is not a proposal for a new media-format or filesystem (like iso9660). Rather, it is a proposal for a simple (a single file) extension to the vast array of existing media formats.

# Specification

## Multimedia Content and Presentation

Multimedia is classified into its content and its presentation.

Multimedia content consists of those non-physical objects that composed the multimedia object.

For example group names, titles, publisher, and publisher date.

As in PML, this multimedia specification attempts to abstract the presentation from the description of multimedia objects [CGDS99 and RHOB].

## The Audio Specification

There are many multimedia types, however, I will start with plain-audio, emphasizing on content descriptions.

Audio is a media format that contains no video or text.

In terms of presentation, audio can be mono, stereo, surround, etc.

## Basic Audio Description Types

An audio object is classified into three basic types:

Music

Voice (intelligible speech)

Samples/(sound effects)

### Voice

This is the simplest form of audio. Voice commonly involves a single speaker, however Multiple speaker dialogs also exist (i.e. a talk radio show)

Voice recordings are useless unless there's some background information with it:

I.e. speaker(s), location, date, recording type (i.e. studio/or live), language

Information about each speaker is also required (i.e. what they look like, their role, what Organizations they represent).

Voice recordings also have genres (i.e. speech, talk show, comedy, classroom, etc).

### Music

Music is a type of audio that represents many genres, for example

Classical, Rock, R&B, Techno, New Age.

Music also needs background (i.e. performers, location, date, recording type (i.e. studio/or live)

Performers in music also have roles, (Vocalist, Bass, Drums. Guitar)

### Samples

Samples are snippets of audio put together for extraction at some later date.

A sound effects record/CD is such an example.

Samples as well have background information, but often more technical:

I.e. (glass shattering inside, plate glass window shattering at 15' away).

In some cases a mix of samples may compose a work of music, or a work of music may just consist of a person's speech. In cases such as this, it is up to the discretion of the content creator to decide what the basic type of the multimedia object is.

## Intended Purpose

Audio can be classified into purpose types:

Educate  
Entertain  
Advertisement  
Historical/Documentary

The intended purpose for an audio object is the stated purpose at the time of creation or recording of the object. For example, the purpose for most music is to simply entertain, whereas a recording of Churchill in 1938 is documentary or historical.

## Temporal Audio Hierarchy

- A Volume contains an Album or set of Albums
- A group of audio recordings lives within an Album
- Audio recordings themselves are separated by Tracks.
- Within an audio stream, indexes may be used to mark sections of an audio track to give additional meaning (i.e. at 3:04 into the song, here's where the drummer drops his sticks), or (at 3:04 into the dialog, Howard Stern begins to speak).

```
<Volume>  
  <Album>  
    <Track>  
      <Index></Index>  
    </Track>  
  </Album>  
</Volume>
```

## Audio Presentation

Audio presentation consists of those attributes that describe the format and playback of an audio file.

Format type (ie wav, mp3,etc)  
Start and End points (for long files)  
Channels (mono, stereo, 4 chan)  
Bitrate (16 bit)  
Frequency (44.1Khz)  
Layer (mpeg layer 3)  
Gap between tracks  
Pre Emphasis  
Volume compensation  
Related images displayed at certain times

There are presentation languages already in existence (for example SMIL [RA98]). However, none were designed to discriminate description from presentation. Also, many such languages are designed explicitly for indexing in a database context (for example MOQL [LOSO97]).

## What tools exists and implementation:

This specification is based on XML using RDF.  
Markup Language: XML [XML98], RDF [RDF99]  
RDF Vocabularies: PICS, Dublin Core, ID3

RDF is an XML framework for describing web and other resources. It includes vocabularies for Dublin Core and a rating system called PICS [PICS98].

The broad goal of RDF is to define a mechanism for describing resources that makes no assumptions about a particular application domain, nor defines (a priori) the semantics of any application domain. The definition of the mechanism should be domain neutral, yet the mechanism should be suitable for describing information about any domain.

The RDF data model provides an abstract, conceptual framework for defining and using metadata. A concrete syntax is also needed for the purposes of creating and exchanging this metadata. This specification of RDF uses the [Extensible Markup Language](#) (XML) encoding as its interchange syntax. RDF also requires the [XML namespace facility](#) to precisely associate each property with the schema that defines the property [RDFS99]. See the RDF spec for details on Schemas and Namespaces.

A single RDF statement seldom appears in isolation; most commonly several properties of a resource will be given together. The RDF XML syntax has been designed to accommodate this easily by grouping multiple statements for the same resource into a `Description` element. The `Description` element names, in an `about` attribute, the resource to which each of the statements apply.

Basic RDF serialization syntax takes the form:

```
[1] RDF          ::= ['<rdf:RDF>'] description* ['</rdf:RDF>']
[2] description ::= '<rdf:Description' idAboutAttr? '>'
propertyElt*
           '</rdf:Description>'
[3] idAboutAttr ::= idAttr | aboutAttr
[4] aboutAttr  ::= 'about="' URI-reference '"'
[5] idAttr     ::= 'ID="' IDsymbol '"'
[6] propertyElt ::= '<' propName '>' value '</' propName '>'
                | '<' propName resourceAttr '/>'
[7] propName   ::= QName
[8] value      ::= description | string
[9] resourceAttr ::= 'resource="' URI-reference '"'
[10] QName     ::= [ NSprefix ':' ] name
[11] URI-reference ::= string, interpreted per [URI]
[12] IDsymbol   ::= (any legal XML name symbol)
[13] name       ::= (any legal XML name symbol)
[14] NSprefix   ::= (any legal XML namespace prefix)
[15] string     ::= (any XML text, with "<", ">", and "&" escaped)
```

The `RDF` element is a simple wrapper that marks the boundaries in an XML document between which the content is explicitly intended to be mappable into an RDF data model instance. The `RDF` element is optional if the content can be known to be RDF from the application context.

`Description` contains the remaining elements that cause the creation of statements in the model instance. The `Description` element may be thought of (for purposes of the basic RDF syntax) as simply a place to hold the identification of the resource being described. Typically there will be more than one statement made about a resource; `Description` provides a way to give the resource name just once for several statements.

When the `about` attribute is specified with `Description`, the statements in the `Description` refer to the resource whose identifier is determined from the `about`. The value of the `about` attribute is interpreted as a URI-reference per Section 4 of [XML]. The corresponding resource identifier is obtained by resolving the URI-reference to absolute form as specified by [XML]. If a fragment identifier is included in the URI-reference then the resource identifier refers only to the subcomponent of the containing resource that is identified by the corresponding fragment id internal to that containing resource (see anchor in

[[Dexter94](#)]), otherwise the identifier refers to the entire resource specified by the URI. A `Description` element without an `about` attribute represents a new resource. Such a resource might be a surrogate, or proxy, for some other physical resource that does not have a recognizable URI. The value of the `ID` attribute of the `Description` element, if present, is the anchor id of this "in-line" resource.

If another `Description` or property value needs to refer to the in-line resource it will use the value of the `ID` of that resource in its own `about` attribute. The `ID` attribute signals the creation of a new resource and the `about` attribute refers to an existing resource; therefore either `ID` or `about` may be specified on `Description` but not both together in the same element. The values for each `ID` attribute must not appear in more than one `ID` attribute within a single document.

A single `Description` may contain more than one *propertyElt* element with the same property name. Each such *propertyElt* adds one arc to the graph. The interpretation of this graph is defined by the schema designer.

Within a *propertyElt*, the `resource` attribute specifies that some other resource is the value of this property; that is, the object of the statement is another resource identified by URI rather than a literal. The resource identifier of the object is obtained by resolving the `resource` attribute URI-reference in the same manner as given above for the `about` attribute. *Strings* must be well-formed XML; the usual XML content quoting and escaping mechanisms may be used if the string contains character sequences (e.g. "<" and "&") that violate the well-formedness rules or that otherwise might look like markup. See [Section 6 of \[RDF\]](#) for additional syntax to specify a property value with well-formed XML content containing markup such that the markup is not interpreted by RDF.

Property names must be associated with a schema. This can be done by qualifying the element names with a namespace prefix to unambiguously connect the property definition with the corresponding RDF schema or by declaring a default namespace as specified in [RDF].

For example:

*Ora Lassila is the creator of the resource <http://www.w3.org/Home/Lassila>.*

is represented in RDF/XML as:

```
<rdf:RDF>
  <rdf:Description about="http://www.w3.org/Home/Lassila">
    <s:Creator>Ora Lassila</s:Creator>
  </rdf:Description>
</rdf:RDF>
```

Here the namespace prefix 's' refers to a specific namespace prefix chosen by the author of this RDF expression and defined in an XML namespace declaration such as

```
:
  xmlns:s=http://description.org/schema/
```

This namespace declaration would typically be included as an XML attribute on the `rdf:RDF` element but may also be included with a particular `Description` element or even an individual *propertyElt* expression. The namespace name URI in the namespace declaration is a globally unique identifier for the particular schema this metadata author is using to define the use of the `Creator` property. Other schemas may also define a property named `Creator` and the two properties will be distinguished via their schema identifiers. Note also that a schema usually defines several properties; a single namespace declaration will suffice to make a large vocabulary of properties available for use.

The complete XML document containing the description above would be:

```
<?xml version="1.0"?>
<rdf:RDF
```

```

xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:s="http://description.org/schema/">
<rdf:Description about="http://www.w3.org/Home/Lassila">
  <s:Creator>Ora Lassila</s:Creator>
</rdf:Description>
</rdf:RDF>

```

Furthermore, namespace declarations can be associated with an individual `Description` element or even an individual `propertyElt` element as in:

```

<?xml version="1.0"?>
<RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <Description about="http://www.w3.org/Home/Lassila">
    <s:Creator xmlns:s="http://description.org/schema/">Ora
Lassila</s:Creator>
  </Description>
</RDF>

```

While the serialization syntax shows the structure of an RDF model most clearly, often it is desirable to use a more compact XML form. The RDF *abbreviated syntax* accomplishes this. As a further benefit, the abbreviated syntax allows documents obeying certain well-structured XML DTDs to be directly interpreted as RDF models.

Three forms of abbreviation are defined for the basic serialization syntax. The first is usable for properties that are not repeated within a `Description` and where the values of those properties are literals. In this case, the properties may be written as XML attributes of the `Description` element. The previous example then becomes:

```

<rdf:RDF>
  <rdf:Description about="http://www.w3.org/Home/Lassila"
                  s:Creator="Ora Lassila" />
</rdf:RDF>

```

Note that since the `Description` element has no other content once the `Creator` property is written in XML attribute form, the XML empty element syntax is employed to elide the `Description` end-tag. Here is another example of the use of this same abbreviation form:

```

<rdf:RDF>
  <rdf:Description about="http://www.w3.org">
    <s:Publisher>World Wide Web Consortium</s:Publisher>
    <s:Title>W3C Home Page</s:Title>
    <s>Date>1998-10-03T02:27</s>Date>
  </rdf:Description>
</rdf:RDF>

```

is equivalent for RDF purposes to

```

<rdf:RDF>
  <rdf:Description about="http://www.w3.org"
                  s:Publisher="World Wide Web Consortium"
                  s:Title="W3C Home Page"
                  s>Date="1998-10-03T02:27" />
</rdf:RDF>

```

Note that while these two RDF expressions are equivalent, they may be treated differently by other processing engines. In particular, if these two expressions were embedded into an HTML document then

the default behavior of a non-RDF-aware browser would be to display the values of the properties in the first case while in the second case there should be no text displayed (or at most a whitespace character). The second RDF abbreviation form works on nested `Description` elements. This abbreviation form can be employed for specific statements when the object of the statement is another resource and the values of any properties given in-line for this second resource are strings. In this case, a similar transformation of XML element names into XML attributes is used: the properties of the resource in the nested `Description` may be written as XML attributes of the `propertyElt` element in which that `Description` was contained.

The second example sentence:

*The individual referred to by employee id 85740 is named Ora Lassila and has the email address lassila@w3.org. The resource <http://www.w3.org/Home/Lassila> was created by this individual.*

is written in RDF/XML using explicit serialization form as

```
<rdf:RDF>
  <rdf:Description about="http://www.w3.org/Home/Lassila">
    <s:Creator rdf:resource="http://www.w3.org/staffId/85740"/>
  </rdf:Description>

  <rdf:Description about="http://www.w3.org/staffId/85740">
    <v:Name>Ora Lassila</v:Name>
    <v:Email>lassila@w3.org</v:Email>
  </rdf:Description>
</rdf:RDF>
```

This form makes it clear to a reader that two separate resources are being described but it is less clear that the second resource is used within the first description. This same expression could be written in the following way to make this relationship more obvious to the human reader. Note that to the machine, there is no difference:

```
<rdf:RDF>
  <rdf:Description about="http://www.w3.org/Home/Lassila">
    <s:Creator>
      <rdf:Description about="http://www.w3.org/staffId/85740">
        <v:Name>Ora Lassila</v:Name>
        <v:Email>lassila@w3.org</v:Email>
      </rdf:Description>
    </s:Creator>
  </rdf:Description>
</rdf:RDF>
```

Using the second basic abbreviation syntax, the inner `Description` element and its contained property expressions can be written as attributes of the `Creator` element:

```
<rdf:RDF>
  <rdf:Description about="http://www.w3.org/Home/Lassila">
    <s:Creator rdf:resource="http://www.w3.org/staffId/85740"
      v:Name="Ora Lassila"
      v:Email="lassila@w3.org" />
  </rdf:Description>
</rdf:RDF>
```

When using this abbreviation form the `about` attribute of the nested `Description` element becomes a resource attribute on the `propertyElt` element, as the resource named by the URI is in both cases the value of the `Creator` property. It is entirely a matter of writer's preference which of the three forms above are used in the RDF source. They all produce the same internal RDF models.

The third basic abbreviation applies to the common case of a `Description` element containing a `type` property (see RDF spec for meaning of `type`). In this case, the resource type defined in the schema corresponding to the value of the `type` property can be used directly as an element name. For example, using the previous RDF fragment if we wanted to add the fact that the resource `http://www.w3.org/staffId/85740` represents an instance of a `Person`, we would write this in full serialization syntax as:

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s="http://description.org/schema/">
  <rdf:Description about="http://www.w3.org/Home/Lassila">
    <s:Creator>
      <rdf:Description about="http://www.w3.org/staffId/85740">
        <rdf:type resource="http://description.org/schema/Person"/>
        <v:Name>Ora Lassila</v:Name>
        <v:Email>lassila@w3.org</v:Email>
      </rdf:Description>
    </s:Creator>
  </rdf:Description>
</rdf:RDF>
```

and using this third abbreviated form as:

```
<rdf:RDF>
  <rdf:Description about="http://www.w3.org/Home/Lassila">
    <s:Creator>
      <s:Person about="http://www.w3.org/staffId/85740">
        <v:Name>Ora Lassila</v:Name>
        <v:Email>lassila@w3.org</v:Email>
      </s:Person>
    </s:Creator>
  </rdf:Description>
</rdf:RDF>
```

While useful, RDF cannot describe sections of audio represented in a stream alone. Separate RDF entries need to be generated for each marked section of audio. Dublin Core [DC98] and PICS [PICS98] do not address multimedia. For this index format, a new/extended vocabulary will be introduced to RDF.

For this specification, XML tags are used to define the presentation structure (or layout) of the audio content. XML tags will also contain attributes representing the object to which it points [GP98]. For representing salient objects within the multimedia object itself, RDF is used. This enables me to abstract the presentation from the format from the description of the content.

### **RDF Schema for audio**

This multimedia XML RDF schema [RDFS99] called *MM* will obviously apply to other formats as well as audio:

```

<rdf:RDF
  xmlns:rdf="http://www.w3.org/TR/WD-rdf-syntax#"
  xmlns:rdfs="http://www.w3.org/TR/WD-rdf-schema#"

<rdf:Description ID="Title">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Title</rdfs:label>
  <rdfs:comment>The name given to the resource, usually by the Creator or
  Publisher.</rdfs:comment> </rdf:Description>

<rdf:Description ID="Creator">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Author/Creator</rdfs:label>
  <rdfs:comment>The person or organization primarily responsible for creating the
  intellectual content of the resource. For example, authors in the case of written
  documents, artists, photographers, or illustrators in the case of visual
  resources.</rdfs:comment></rdf:Description>

  ** Note, these person tags apply to creators and contributors ***
  <rdf:Person ID="Role">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Role </rdfs:label>
  <rdfs:comment>The person's role in the creation/production of the work. For
  example, lead vocals.</rdfs:comment></rdf:Person>

  <rdf:Person ID="Name">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Name</rdfs:label>
  <rdfs:comment>The person's full name</rdfs:comment></rdf:Person>

  <rdf:Person ID="Age">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Age</rdfs:label>
  <rdfs:comment>The person's age</rdfs:comment></rdf:Person>

  <rdf:Person ID="Sex">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Sex</rdfs:label>
  <rdfs:comment>The person's sex (m or f)</rdfs:comment></rdf:Person>

  <rdf:Person ID="Image">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Image</rdfs:label>
  <rdfs:comment>References a URI to a gif or jpeg image of the
  person</rdfs:comment></rdf:Person>

  <rdf:Person ID="Clip">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Clip</rdfs:label>
  <rdfs:comment>References a URI to a short mpeg video clip of the
  person</rdfs:comment></rdf:Person>

<rdf:Description ID="Subject">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Subject</rdfs:label>
  <rdfs:comment>The topic of the resource. Typically, subject will be expressed as
  keywords or phrases that describe the subject or content of the resource. The use
  of controlled vocabularies and formal classification schemes is encouraged.

  For audio, subjects in music are esoteric, however in voice and samples, these are
  more concrete.</rdfs:comment></rdf:Description>

<rdf:Description ID="Description">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Description</rdfs:label>
  <rdfs:comment> A textual description of the content of the resource, including
  abstracts in the case of document-like objects or content descriptions in the case
  of visual resources.
  For audio samples, for example: Glass shattering at 15'
  away.</rdfs:comment></rdf:Description>

```

```

<rdf:Description ID="Publisher">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Publisher</rdfs:label>
  <rdfs:comment>The entity responsible for making the resource available in its
  present form, such as a publishing house, a university department, or a corporate
  entity.</rdfs:comment></rdf:Description>

  <rdf:Publisher ID="Logo">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Logo</rdfs:label>
  <rdfs:comment>This entity references a Corporate logo for the publisher. The
  syntax is a standard URI reference to a GIF or JPEG image
  </rdfs:comment></rdf:Description>

  <rdf:Publisher ID="Name">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Name</rdfs:label>
  <rdfs:comment>The name of the publisher responsible for making the resource
  available in its present form, such as a publishing house, a university
  department, or a corporate entity.</rdfs:comment></rdf:Description>

  <rdf:Publisher ID="Role">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Role</rdfs:label>
  <rdfs:comment>Specifies the role the Publisher played when creating the publishing
  the work.</rdfs:comment></rdf:Description>

  <rdf:Publisher ID="Location">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Location</rdfs:label>
  <rdfs:comment>Specifies the location (address,city,state) of the
  publisher.</rdfs:comment></rdf:Description>

<rdf:Description ID="Contributor">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Other Contributors</rdfs:label>
  <rdfs:comment>A person or organization not specified in a Creator element who has
  made significant intellectual contributions to the resource but whose contribution
  is secondary to any person or organization specified in a Creator element (for
  example, editor, transcriber, and illustrator).</rdfs:comment> </rdf:Description>

<rdf:Description ID="Date">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Date</rdfs:label>
  <rdfs:comment>A date associated with the creation or availability of the resource.
  Such a date is not to be confused with one belonging in the Coverage element,
  which would be associated with the resource only insofar as the intellectual
  content is somehow about that date. Recommended best practice is defined in a
  profile of ISO 8601 [Date and Time Formats (based on ISO8601), W3C Technical Note,
  http://www.w3.org/TR/NOTE-datettime] that includes (among others) dates of the
  forms YYYY and YYYY-MM-DD. In this scheme, for example, the date 1994-11-05
  corresponds to November 5, 1994.</rdfs:comment></rdf:Description>

<rdf:Description ID="Type">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Type</rdfs:label>
  <rdfs:comment>The category of the resource. For audio, this represents one of the
  basic types (voice, music, or samples).</rdfs:comment>
  </rdf:Description>

<rdf:Description ID="Identifier">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Identifier</rdfs:label>
  <rdfs:comment>A string or number used to uniquely identify the resource. Examples
  for networked resources include URLs and URNs (when implemented). Other globally-
  unique identifiers, such as International Standard Book Numbers (ISBN) or other
  formal names are also candidates for this
  element.</rdfs:comment></rdf:Description>

```

NOTE: Identifier itself is too vague, entries for serial numbers, catalog numbers, as well as publishing identifiers are provided below:

```
<rdf:Identifier ID="SerialID">
<rdf:Identifier ID="CatalogID">
<rdf:Identifier ID="PublishID">

<rdf:Description ID="Language">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Language</rdfs:label>
  <rdfs:comment>The language of the intellectual content of the resource. Where
  practical, the content of this field should coincide with RFC 1766 [Tags for the
  Identification of Languages, http://ds.internic.net/rfc/rfc1766.txt ]; examples
  include en, de, es, fi, fr, ja, th, and zh.</rdfs:comment></rdf:Description>
  (character set)

<rdf:Description ID="Relation">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Relation</rdfs:label>
  <rdfs:comment>An identifier of a second resource and its relationship to the
  present resource. This element permits links between related resources and
  resource descriptions to be indicated. Examples include an edition of a work
  (IsVersionOf), a translation of a work (IsBasedOn), a chapter of a book
  (IsPartOf), and a mechanical transformation of a dataset into an image
  (IsFormatOf). For the sake of interoperability, relationships should be selected
  from an enumerated list that is currently under development in the workshop
  series.</rdfs:comment> </rdf:Description>

<rdf:Description ID="Coverage">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Coverage</rdfs:label>
  <rdfs:comment>The spatial or temporal characteristics of the intellectual content
  of the resource. Spatial coverage refers to a physical region (e.g., celestial
  sector); use coordinates (e.g., longitude and latitude) or place names that are
  from a controlled list or are fully spelled out. Temporal coverage refers to what
  the resource is about rather than when it was created or made available (the
  latter belonging in the Date element); use the same date/time format (often a
  range) [Date and Time Formats (based on ISO8601), W3C Technical Note,
  http://www.w3.org/TR/NOTE-datettime] as recommended for the Date element or time
  periods that are from a controlled list or are fully spelled out.

  For audio, this tag has little meaning. It is usefull for images and video
  contexts.</rdfs:comment></rdf:Description>

<rdf:Description ID="Rights">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Rights</rdfs:label>
  <rdfs:comment>A rights management statement, an identifier that links to a rights
  management statement, or an identifier that links to a service providing
  information about rights management for the resource.</rdfs:comment>
  </rdf:Description>

  <rdf:Rights ID="Warning">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Warning</rdfs:label>
  <rdfs:comment>A warning statement about the copy status of the intellectual
  property. (ie FBI warning on video tapes).</rdfs:comment> </rdf:Description>

  <rdf:Rights ID="Copyright">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Copyright</rdfs:label>
  <rdfs:comment>A statement about the copyright (ie © 1998, My
  Corporation)</rdfs:comment> </rdf:Description>

  <rdf:Rights ID="Printright">
  <rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
  <rdfs:label>Printright</rdfs:label>
  <rdfs:comment>A statement about the printright (ie (p) 1998, My
  Corporation)</rdfs:comment> </rdf:Description>
```

```

<rdf:Rights ID="Patent">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Patent</rdfs:label>
<rdfs:comment>A statement about the patent held on this intellectual property (ie
US Patent #122331 issued 1-1-1998)</rdfs:comment> </rdf:Description>

<rdf:Rights ID="Owner">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Owner</rdfs:label>
<rdfs:comment>A statement about the owner of the intellectual property
(ie, My Corporation, All rights reserved).</rdfs:comment> </rdf:Description>

<rdf:Description ID="Purpose ">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Purpose</rdfs:label>
<rdfs:comment>A statement of the intended purpose of this resource. Purposes, for
example, can be to educate, entertain, market, or have historical
significance.</rdfs:comment> </rdf:Description>

<rdf:Description ID="Genre">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Genre</rdfs:label>
<rdfs:comment>A statement of the genre of this resource. Genre types are listed
below</rdfs:comment> </rdf:Description>

<rdf:Description ID="Location">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Location</rdfs:label>
<rdfs:comment>A statement of where the resource was created. For example, the
location of where a speech was recorded.</rdfs:comment> </rdf:Description>

** these also apply to a publisher's location **
<rdf:Location ID="Country">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Country</rdfs:label>
</rdf:Description>

<rdf:Location ID="State">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>State</rdfs:label>
</rdf:Description>

<rdf:Location ID="Province">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Province</rdfs:label>
</rdf:Description>

<rdf:Location ID="Address">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Address</rdfs:label>
</rdf:Description>

<rdf:Location ID="City">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>City</rdfs:label>
</rdf:Description>

<rdf:Description ID="CreationType">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>CreationType</rdfs:label>
<rdfs:comment>Type of creation used when the resource was created. For audio,
this refers to Live or Studio recordings</rdfs:comment> </rdf:Description>

<rdf:Description ID="Role">
<rdf:type rdf:resource="http://www.w3.org/TR/WD-rdf-syntax#Property"/>
<rdfs:label>Role</rdfs:label>
<rdfs:comment>Role of the current object. For audio, this might include a short
description of a performer, for example Vocalist, Guitar, Drums</rdfs:comment>
</rdf:Description>

</rdf:RDF>

```

Note, the PICS rating system was not presented here, However, it is also a part of this vocabulary specification. PICS is a work-in-progress specification [PICS98].

## Examples

Example of an album with 2 tracks. Track 1 has 2 indexes, Track2 has one index.

```
<?xml version="1.0"?>
```

```
<Album>
```

```
<RDF xmlns="http://w3.org/TR/1999/PR-rdf-syntax-19990105#"
  xmlns:MM="http://www.ccs.neu.edu/home/bchafy/mmns">
  <Bag ID="tracks">
    <li resource="T1" />
    <li resource="T2" />
  </Bag>
  <Description about="tracks">
    <MM:Publisher>Big company studios</MM:Publisher>
    <MM:Rights>All Rights Reserved © 1997</MM:Rights>
    <MM:Date>1998-08-08.15:00:00</MM:Date>
    <MM:Titie>The big Album</MM:Title>
    <MM:Creator>Fiona Apple</MM:Creator>
  </Description>
</RDF>
```

```
<Track ID="T1" href="file:/music/songs/blasong.wav">
```

```
  <Index start="03:02:00"/>
```

```
  <Index start="05:21:00"/>
```

```
</Track>
```

```
<Track ID="T2" href="file:/music/songs/blasong2.mp3">
```

```
<RDF xmlns="http://w3.org/TR/1999/PR-rdf-syntax-19990105#"
  xmlns:MM="http://www.ccs.neu.edu/home/bchafy/mmns">
```

```
  <Description about="this">
    <MM:Titie>My good song</MM:Title>
    <MM:Type>Music</MM:Type>
    <MM:Genre>Pop</MM:Genre>
    <MM:SourceType>Live</MM:SourceType>
    <MM:Location>Dublin Core Studios, UK</MM:Location>

    <MM:Creator>
      Bryan Chafy
      <MM:Role>Lead Vocals</MM:Role>
    </MM:Creator>
    <MM:Contributor>
      Fiona Apple
      <MM:Role>Drums</MM:Role>
    </MM:Contributor>
  </Description>
```

```
  <Index start="02:00:00">
```

```
    <RDF xmlns="http://w3.org/TR/1999/PR-rdf-syntax-19990105#"
      xmlns:MM="http://www.ccs.neu.edu/home/bchafy/mmns">
```

```
      <Description about="this">
```

```
        <MM:Description>Here's wher the drummer loses his sticks</MM:Description>
```

```
      </Description>
```

```
    </Index>
```

```
</Track>
```

```
</Album>
```

Although useful, RDF is a completely optional parameter within the structure of Albums, Tracks, etc. For example, the following is also valid markup:

```
<Album>
<Track href="file1.wav">
  <Index start="03:02:00"/>
  <Index start="05:21:00"/>
</Track href="file2.mp3">
<Track>
  <Index start="02:00:00"/>
</Track>
</Album>
```

However, it says nothing about the content of the data, so it is not recommended.

### XML DTD for Audio

#### Album Element and Attribute:

```
<! ATTLIST album id "CDATA" "0">
<!ELEMENT album - - (cover?, RDF?, (image|track|textdata)+)
```

#### Index Element and Attributes:

```
<!ATTLIST index START CDATA #REQUIRED
                END CDATA #REQUIRED>
<!ELEMENT index - - (RDF?)>
```

#### Track Element and Attributes:

```
<!ATTLIST track
  HREF CDATA #REQUIRED
  TYPE CDATA #IMPLIED
  ID CDATA #IMPLIED
  START CDATA #IMPLIED
  END CDATA #IMPLIED
  GAP NUMBER 2
  CHANNELS NUMBER #IMPLIED
  BITRATE NUMBER #IMPLIED
  FREQUENCY NUMBER #IMPLIED
  LAYER NUMBER #IMPLIED
  SIZE NUMBER #IMPLIED
  FORMAT CDATA #IMPLIED
  ASPECT CDATA #IMPLIED
  FRAMERATE CDATA #IMPLIED
  PREEMPHASIS CDATA #IMPLIED
  PREGAIN CDATA #IMPLIED >

<!ELEMENT track - - (RDF?,link*,index*,streamimage*,streamtextdata*)>
```

### Genre Types

Genre is ever-changing. A genre may be re-classified many times over the course of the multimedia object's life. However, adhering to the original stated genre and avoiding the use of genres such as "retro" and "oldies" is highly recommended.

### **Voice Genre Types:**

Speech  
Conference  
Surveillance  
Talk Show  
Comedy  
Educational  
Political

### **Music Genre types (as defined by the ID3 specification):**

Genres(0) = "Blues"  
Genres(1) = "Classic Rock"  
Genres(2) = "Country"  
Genres(3) = "Dance"  
Genres(4) = "Disco"  
Genres(5) = "Funk"  
Genres(6) = "Grunge"  
Genres(7) = "Hip-Hop"  
Genres(8) = "Jazz"  
Genres(9) = "Metal"  
Genres(10) = "New Age"  
Genres(11) = "Oldies"  
Genres(12) = "Other"  
Genres(13) = "Pop"  
Genres(14) = "R&B"  
Genres(15) = "Rap"  
Genres(16) = "Reggae"  
Genres(17) = "Rock"  
Genres(18) = "Techno"  
Genres(19) = "Industrial"  
Genres(20) = "Alternative"  
Genres(21) = "Ska"  
Genres(22) = "Death Metal"  
Genres(23) = "Pranks"  
Genres(24) = "Soundtrack"  
Genres(25) = "Euro-Techno"  
Genres(26) = "Ambient"  
Genres(27) = "Trip-Hop"  
Genres(28) = "Vocal"  
Genres(29) = "Jazz Funk"  
Genres(30) = "Fusion"  
Genres(31) = "Trance"  
Genres(32) = "Classical"  
Genres(33) = "Instrumental"  
Genres(34) = "Acid"  
Genres(35) = "House"  
Genres(36) = "Game"  
Genres(37) = "Sound Clip"  
Genres(38) = "Gospel"  
Genres(39) = "Noise"  
Genres(40) = "Alternative Rock"    Genres(41) = "Bass"  
Genres(42) = "Soul"    Genres(43) = "Punk"    Genres(44) = "Space"

Genres(45) = "Meditative" Genres(46) = "Instrumental Pop"  
Genres(47) = "Instrumental Rock" Genres(48) = "Ethnic"  
Genres(49) = "Gothic" Genres(50) = "Darkwave"  
Genres(51) = "Techno-Industrial" Genres(52) = "Electronic"  
Genres(53) = "Pop-Folk" Genres(54) = "Eurodance" Genres(55) = "Dream"  
Genres(56) = "Southern Rock" Genres(57) = "Comedy" Genres(58) = "Cult"  
Genres(59) = "Gangsta" Genres(60) = "Top 40"  
Genres(61) = "Christian Rap" Genres(62) = "Pop/Funk"  
Genres(63) = "Jungle" Genres(64) = "Native American"  
Genres(65) = "Cabaret" Genres(66) = "New Wave"  
Genres(67) = "Psychadelic" Genres(68) = "Rave"  
Genres(69) = "Showtunes" Genres(70) = "Trailer" Genres(71) = "Lo-Fi"  
Genres(72) = "Tribal" Genres(73) = "Acid Punk"  
Genres(74) = "Acid Jazz" Genres(75) = "Polka" Genres(76) = "Retro"  
Genres(77) = "Musical" Genres(78) = "Rock & Roll"  
Genres(79) = "Hard Rock" Genres(80) = "Folk" Genres(81) = "Folk/Rock"  
Genres(82) = "National Folk" Genres(83) = "Swing"  
Genres(84) = "Fast Fusion" Genres(85) = "Bebob" Genres(86) = "Latin"  
Genres(87) = "Revival" Genres(88) = "Celtic" Genres(89) = "Bluegrass"  
Genres(90) = "Avantgarde" Genres(91) = "Gothic Rock"  
Genres(92) = "Progressive Rock" Genres(93) = "Psychedelic Rock"  
Genres(94) = "Symphonic Rock" Genres(95) = "Slow Rock"  
Genres(96) = "Big Band" Genres(97) = "Chorus"  
Genres(98) = "Easy Listening" Genres(99) = "Acoustic"  
Genres(100) = "Humour" Genres(101) = "Speech" Genres(102) = "Chanson"  
Genres(103) = "Opera" Genres(104) = "Chamber Music"  
Genres(105) = "Sonata" Genres(106) = "Symphony"  
Genres(107) = "Booty Bass" Genres(108) = "Primus"  
Genres(109) = "Porn Groove" Genres(110) = "Satire"  
Genres(111) = "Slow Jam" Genres(112) = "Club" Genres(113) = "Tango"  
Genres(114) = "Samba" Genres(115) = "Folklore" Genres(116) = "Ballad"  
Genres(117) = "Power Ballad" Genres(118) = "Rhythmic Soul"  
Genres(119) = "Freestyle" Genres(120) = "Duet"  
Genres(121) = "Punk Rock" Genres(122) = "Drum Solo"  
Genres(123) = "A Capella" Genres(124) = "Euro-House"  
Genres(125) = "Dance Hall" Genres(126) = "Goa"  
Genres(127) = "Drum & Bass" Genres(128) = "Club-House"  
Genres(129) = "Hardcore" Genres(130) = "Terror" Genres(131) = "Indie"  
Genres(132) = "BritPop" Genres(133) = "Negerpunk"  
Genres(134) = "Polsk Punk" Genres(135) = "Beat"  
Genres(136) = "Christian Gangsta Rap" Genres(137) = "Heavy Metal"  
Genres(138) = "Black Metal" Genres(139) = "Crossover"  
Genres(140) = "Contemporary Christian" Genres(141) = "Christian Rock"  
Genres(142) = "Merengue" Genres(143) = "Salsa"  
Genres(144) = "Thrash Metal" Genres(145) = "Anime"  
Genres(146) = "JPop" Genres(147) = "Synthpop"

## **Issues**

When a multimedia file contains provisions for tagged descriptions, there is a risk of keeping a redundancy of information.

### **Advantages**

- When the file is copied, its descriptions live with it

### **Disadvantages**

- The index also contains the information, making some or all of the information redundant
- Since there is no standard for file markup, different files will have allotments for different tags and descriptions. The XML index may contain more descriptive and presentation information than the file's markup can handle. (ID3v1 is an example of this).

### **Problem**

How to avoid losing index data when a file is moved from a media with an index to a media with no index?

### **Solutions (none are good)**

- When a file is copied, move as much description information as possible from the index into the file's description feature, risking losing some information.
- Package the tagged description from the index into a separate file (with same name as the resource, but with a different extension). There will be two files to maintain.
- Use resource forks. Not all operating systems support this feature.

## **Continuing Work With Video and Other Formats**

There is continuing work-in-progress for supporting streaming video as well as other media formats (i.e. video game ROM formats). The audio specification should easily port to these formats with little trouble. An initial design goal of this specification is to keep the RDF vocabulary as descriptive (and yet small) as possible. To avoid the metadata equivalent of spaghetti code [PBCCM99], this RDF schema was designed to accommodate audio as well as video and other formats.

A presentation tool is needed. This will of course vary from context to context. However, in a PC environment, the tool can be as simple as a web browser. This is a work-in-progress.

Side notes about video:

Video is composed of tracks and indexes as well.

Video can be annotated (i.e. text overlay). This may go beyond what an index can do however.

Video zoom, pan, color adjustment, are examples of attributes.

### **Context for this Project**

This project is the result of readings and work done during the winter quarter of 1998/1999 at Northeastern University. The initial idea and proposal for this project was conceived by Bryan Chafy during the Spring of 1998.

Special thanks to Professor Robert Futrelle PhD at Northeastern University for his feedback and mentoring during this project.

## References

[RDFS99]

*Resource Description Framework  
(RDF) Schema Specification*  
<http://www.w3.org/TR/PR-rdf-schema/>

*W3C Proposed Recommendation 03 March 1999*

[RDF99]

*Resource Description Framework  
(RDF) Model and Syntax Specification*  
<http://www.w3.org/TR/REC-rdf-syntax/>

*W3C Recommendation 22 February 1999*

[PICS98]

*Platform for Internet Content Selection (PICS)*  
<http://www.w3.org/PICS/>  
*Last updated: June 1, 1998*

[XML98]

*Extensible Markup Language (XML<sup>TM</sup>)*  
<http://www.w3.org/XML/>  
*The current W3C Recommendation is [XML 1.0](#), February 1998*

[MNID3]

*ID3*  
<http://www.id3.org/>  
*Copyright © 1998 Martin Nilsson*  
*Last Updated: February 2, 1999*

[DC98]

*Dublin Core Metadata Initiative*  
*Dublin Core Metadata Element Set: Reference Description*  
<http://purl.oclc.org/dc/>  
*Dublin Core user Guide Working Draft 1998-07-31*

[GP98]

*Charles Goldfarb, Paul Prescod*  
*The XML Handbook*  
*Part 5 (The Technology of XML pg. 429-629)*  
*Prentice Hall Publishing, 1998*

[LOS097]

J. Z. Li, M.T. Özsu, D. Szafron, V. Oria. "[MOQL: A Multimedia Object Query Language](#)", *The Third International Workshop on Multimedia Information Systems*, Como, Italy, September 1997, pg. 19-28.

[MSXML99]

*XML (Extensible Markup Language)*  
*Microsoft Worksop on XML For MSIE 5.0*  
<http://www.microsoft.com/workshop/c-frame.htm#/xml/default.asp>  
*© 1999 Microsoft Corporation. All rights reserved. Terms of use.*

[ PBCCM99 ]

*Using Distributed Objects to build the Stanford Digital Library Infobus*  
Andreas Paepcke, Michelle Baldinado, Chen-Chuan Chang, Steve Cousins, Hector Molina  
*IEEE Computer Society Magazine*  
February, 1999 pg. 80-87

[CGDS99]

Ram, Ashwin, Catrambone, Richard, Guzdial, Mark, Kehoe, Collen, McCrickard, D. and Stasko, John, "  
[PML: Representing Procedural Domains for Multimedia Presentations](#)" Graphics, Visualization, and  
Usability Center, Georgia Institute of Technology, Atlanta, GA, Technical Report GIT-GVU-98-20,  
August 1998.  
To appear in *IEEE Multimedia* (1999).

[RHOB]

*Structural Distinctions between Hypermedia Storage and Presentation*  
Lloyd Rutledge, Lynda Hardman, Jacco van Ossenbruggen, Dick Bulterman  
*Proceedings of ACM Multimedia 98*, September 1998.

[RA98]

*Synchronized multimedia for the WWW*  
Franck Rousseau and Andrzej Duda.  
*Computer Networks and ISDN Systems*, pages 417--429, April 1998.

## **Other Papers Consulted / Readings:**

*MPEG-7 Seminar Multimedia Content Description Interface*  
Bristol, 9<sup>th</sup> April 1997

*Multimedia Databases and MPEG-7*  
*Call for Papers*  
*IEE Colloquium*  
IEE Savoy Place, London UK January 29, 1999

*MPEG-7: Context and Objectives*  
<http://drogo.cselt.stet.it/mpeg/standards/mpeg-7/mpeg-7.htm>  
version - 10  
October 1998 / Atlantic City, USA

*A Glossary of Digital Library Standards, Protocols and Formats*  
Suzan Haigh  
Network Notes #54  
ISSN 1201-4338  
Information Technology Services  
National Library of Canada May 6, 1998

*Evaluating Object DBMSs for Multimedia*  
Paul Pazandak, Jaideep Srivastava  
*IEEE Multimedia* pg 34 - 49  
July-September 1997

*Metadata Visualization for Digital Libraries*  
*Interactive Timeline Editing and Review*  
Vijoy Kumar, Richard Futura, Robert Allen  
*DL '98 Proceedings of the 3<sup>rd</sup> ACM Conference on Digital Libraries* pg 126-133

*IMMPS: A Multimedia Presentation Design System*  
Timothy Shith, Ruth Davis  
*IEEE Multimedia* pg 67-78  
April-June 1997

*An Object-Oriented SGML/HyTime Compliant Multimedia Database Management System*  
*Proceedings of the conference on Multimedia '97* pg. 239-249

*A framework for Multimedia Database Systems*  
4<sup>th</sup> African Conference in Computer Science  
Dakar, Senegal, October 1998 pg.293-304

*Multimedia Database Management-Requirements and Issues*  
Donald Adjeroh, Kingsley C Nwosu  
*IEEE Multimedia* pg 24-33  
July-September 1997

*The Internet, Intranets, and the AI Renaissance*  
Danial E O'Leary  
*IEEE Computer* pg 71-78  
January 1997

*Design Patterns for Interactive Musical Systems*

*Jan Borchers, Max Muhlhauser*

*IEEE Multimedia* pg 36 – 46

*July-September 1998*

*Content-Based Indexing of Multimedia Databases*

*Jian-Kang Wu*

*IEEE Transactions on Knowledge and Data Engineering, Vol 9, No 6* pg 978 - 989

*November, December 1997*

*The Asilomar Report on Database Research*

*Government web site*

*September 1998*

*Managing multimedia information with parallel systems*

*Agry Krikelis*

*IEEE Concurrency* pg 2-6

*April-June 1998*