PRISM:

Publishing Requirements for Industry Standard Metadata

PRISM Specification: Modular: Version 1.3

The PRISM Rights Language Namespace

2005 10 01

Copyright and Legal Notices

Table of Contents

1 Status

1.1 Document Status

The status of this document is:

✓	Draft
✓	Released for Public Comment
✓	Released

1.2 Document Location

The location of this document is:

http://www.prismstandard.org/ specifications/1.3//PRISM_rights_namespace_1.3.pdf

1.3 Version History

Version Number	***Release Date***	***Editor***	***Description***
1.2	1/26/05	McConnell	Converted from unmodularized PRISM spec v 1.2
1.3A	6/17/05	Kennedy	Clarify element defs and examples, Add RDF discussion
1.3B	7/13/05	Kennedy	Resolve group comments
Final	10/01/0-5	Kennedy	Resolve Industry Comments

2 PRISM Documentation Structure

As of this release, PRISM is described in a set of formal, modularized documents that, taken together, represent "the PRISM Specification." Together these documents comprise the PRISM Documentation Package.

The initial release of the modularized PRISM Documentation Package, is the equivalent of the single document PRISM 1.2 Specification that was approved in December 2004. Moving forward, the monolithic PRISM Specification will no longer be maintained. All revisions will be made to individual documents in the PRISM Documentation Package, with each being versioned separately. Over time, new documents may also be added to the documentation set that makes up the PRISM Specification..

2.1 Normative and Non-normative Sections

Documents in the PRISM Documentation Package may contain both normative and non-normative material; normative material describes element names, attributes, formats, and the contents of elements that is required in order for content or systems to comply with the PRISM Specification. Non-normative material explains, expands on, or clarifies the normative material, but it does not represent requirements for compliance. Normative material in the PRISM Documentation Package is explicitly identified as such; any material not identified as normative can be assumed to be non-normative.

2.1.1 Requirement Wording Note

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC-2119]. The PRISM Specification also uses the normative term, "STRONGLY ENCOURAGES," which should be understood as a requirement equivalent to MUST in all but the most extraordinary circumstances.

Capitalization is significant; lower-case uses of the key words are intended to be interpreted in their normal, informal, English language way.

2.2 The PRISM Documentation Package

The PRISM Documentation Package consists of:

Document	***Description***
PRISM Introduction [PRISMINT]	Overview, background, purpose and scope of PRISM; examples; contains no normative material.
PRISM Compliance [PRISMCOMP]	Describes two profiles of PRISM compliance for content and systems; includes normative material.
The PRISM Namespace [PRISMPRISMNS]	Describes the elements contained in the PRISM namespace; includes normative material.
The PRISM Subset of the Dublin Core Namespace [PRISMDCNS]	Describes the elements from the Dublin Core namespace that are included in PRISM; includes normative material.
The PRISM Rights Language Namespace [PRISMRLNS]	Describes the elements contained in the PRISM Rights Language Namespace; includes normative material.
The PRISM Inline Markup Namespace [PRISMIMNS]	Describes the elements contained in the PRISM Inline Markup Namespace; includes normative material.
The PRISM Controlled Vocabulary Namespace [PRISMCVNS]	Describes the elements contained in the PRISM Controlled Vocabulary Namespace; includes normative material.
The PRISM Aggregator Message Namespace [PRISMAMNS]	Describes the elements contained in the PRISM Aggregator Message Namespace; includes normative material.

Table 1.0: PRISM Documentation Package

2.2.1 Additional PRISM Documentation

The PRISM Aggregator Message (PAM), a DTD-based application of PRISM, adds a small namespace of its own, formally described in [PRISMAMNS]. The structure and use of PAM are described separately in Guide to the PRISM Aggregator Document Type Definition (DTD) V. 1.1. [PAMGUIDE]

2.2.2 Access to PRISM Documentation

The PRISM Documentation Package, the PAM Guide (see above), the PAM DTD, and a range of other information concerning PRISM are all publicly and freely available on the PRISM website, www.prismstandard.org.

3 Introduction

3.1 Purpose and Scope

The purpose of this document is to describe the basic metadata elements that the PRISM Working Group has defined and included in the PRISM Rights Language namespace. All of section 4 of this document is normative.

Note: *This document describes element models and provides examples for profile 2 PRISM Compliance [PRISMCOMP] that documents encoding content using the PRISM Profile of RDF. Profile 1 PRISM (well formed XML, with no requirement for RDF), is described separately in Guide to the PRISM Aggregator Document Type Definition (DTD) V. 1.1. [PAMGUIDE].*

3.2 About RDF

The Resource Description Framework (RDF) is a language for representing information about resources in the World Wide Web but can be used to represent information about any resource that can be identified with a URI, or *Uniform Resource Identifier*. It is particularly useful for representing metadata about resources, such as the title, author modification date of a digital asset and copyright and licensing information for a resource. RDF describes resources in terms of simple properties and property values.

The underlying structure of any expression in RDF is a triple consisting of a Subject, a Predicate and an Object. A set of such triples is called an RDF graph. Figure 1 shows a node and directed-arc diagram of a single triple.

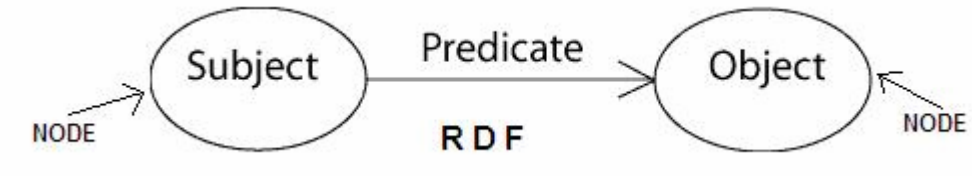

Figure 1.0 RDF Graph

The Predicate specifies a characteristic or property of the Subject. The Object provides the value for the property. For example: The "Big Book of Poems" was authored/created by "D. Kennedy". Here the Subject is the Big Book of Poems. The Predicate or Property we are describing is "was created by" and the value of the property, or Object, is "D. Kennedy".

Figure 2.0 Sample RDF Graph

Subject nodes and predicates must be URIs. An object node may be a URI reference, a literal, or blank (having no separate form of object identification itself).

When expressing RDF in XML, we express the nodes, properties and property values with XML elements and attributes. When using XML to represent RDF triples, there is far greater flexibility in

tagging than we are used to when we define XML elements and attributes with an XML DTD. RDF is designed to represent information in a minimally constraining, flexible way. The impact of combining XML with RDF is that several XML representation models can exist for the same RDF Graph. In other words, the content model and attributes can vary in a way that is not easy to define using an XML DTD. This is a bit foreign to those from a strict XML world where elements have one fixed content model and attribute definition. And it makes writing documentation for XML/RDF elements and attributes quite challenging.

Consider the following options that RDF offers when expressed as XML:

3.3 Specifying Nodes

A URI reference, a literal, or a hierarchy of elements can be used to indicate what a node represents or is used to give the node a value. The tagging of the graph in XML differs depending upon our model for providing Node property values:

```
<dc:description>Browse our catalog of desktop and
notebook computers to find one just right for you.</dc:description>
```

Example 1.0 Literal provides Node value for the dc:description property

```
<dc:description rdf:resource=
"http://www2.rhbnc.ac.uk/Music/Archive/Disserts/attinell.html"/>
```

Example 2.0 URI Reference provides Node value for the dc:description property

3.3.1 URI References

RDF allows property values to be represented by a literal or by a URI. Each representation has different characteristics, so it is important to know about those characteristics in order to make the right design choice. The advantage of URIs over literals is their lack of ambiguity. Literals however are often simpler and more convenient. But either option is valid and is documented in the PRISM Specification.

3.3.2 Literals

To complicate matters even further, there are different types of literals in RDF. These must be coded differently in XML and an RDF processor will handle them differently. To start with, literals may be *plain* or *typed*:

3.3.2.1 Plain Literals

A **plain literal** is a string combined with an optional language tag (xml:lang). This may be used for plain text in a natural language. As recommended in the RDF formal semantics [RDF-SEMANTICS], these plain literals are self-denoting. This means that we do not have to specify a plain literal to an RDF processor; it simply assumes it is dealing with this literal type.

3.3.2.2 Typed Literal

A **typed literal** is a string combined with a datatype URI. It denotes the member of the identified datatype's value space obtained by applying the lexical-to-value mapping to the literal string.

Datatypes are used by RDF in the representation of values such as integers, floating point numbers and dates. There is no built-in concept of numbers or dates or other common values in RDF. Rather, RDF

defers to datatypes that are defined separately, and identified with URI references. The predefined XML Schema datatypes [XML-SCHEMA2] are widely used for this purpose.

```
    <rdf:Description rdf:about="story.xml">
          <prism:embargoDate rdf:datatype="http://www.w3.org/TR/2004/REC-xmlschema-2-
20041028/#dateTime">2001-03-09:00:00:01</prism:embargoDate>
    </rdf:Description>
```

Example 3.0 Typed literal for prism:embargoDate

3.3.2.3 XML Literals

Some literals contain XML markup. **XML literals** is a string combined with a rdf:parseType="literal" attribute that indicates a fragment of XML is embedded. This signals the RDF processor to handle the literal as an XML fragment.

```
<dc:description rdf:parseType="Literal">
 Describes the infamous criminal and gunfighter,
 <em>Billy the Kid</em>.
</dc:description>
```

Example 4.0 Using an XML Literal

3.3.3 Nodes Made of Elements

A third kind of node is known as a **blank node**. This is a node that does not have properties specified with a URI or a literal, but is made up of other elements that themselves have properties.

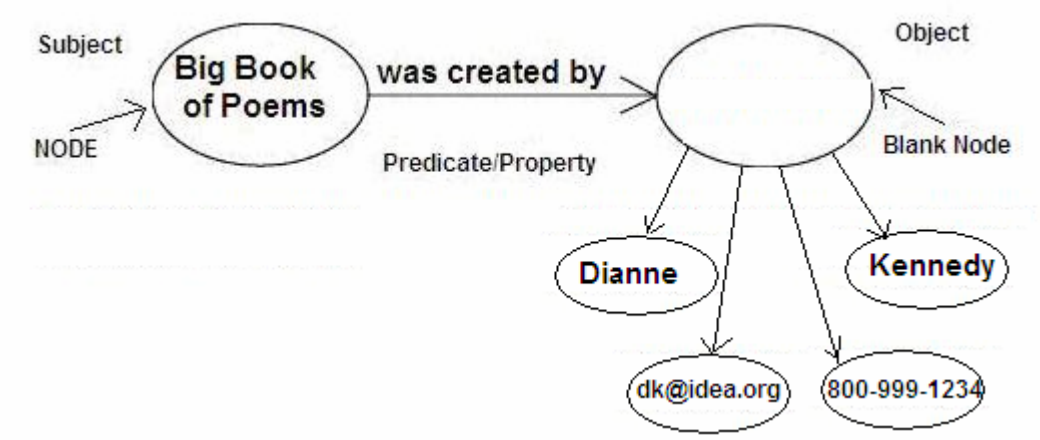

Figure 3.0 RDF Graph with a Blank Node

A blank node must have the `rdf:parseType="Resource"` attribute on the containing property element to turn the property element into a property-and-node element, which can itself have both property elements and property attributes.

```
<dc:rights rdf:parseType="Resource">
  <prism:expirationDate>2001-04-09</prism:expirationDate>
  <prism:embargoDate>2001-05-09</prism:embargoDate>
</dc:rights>
```

Example 5.0 Blank Node with rdf:parseType="Resource" attribute

3.4 Grouped Property Values

There is often a need to describe **groups** of things as a property value. If the "Big Poetry Book" was created by several authors, how could we indicate that? RDF provides several predefined (built-in) types and properties that can be used to describe a group of property values. XMP [XMP] uses these mechanisms when multiple field values are to be entered. If there are multiple values for a metadata field for the resource PRISM recommends listing the multiple values inside a single PRISM element using the RDF Bag, Alt or Seq containers to be compatible with XMP.

First, RDF provides a **container vocabulary** consisting of three predefined types (together with some associated predefined properties). A **container** is a resource that contains a group of values. Containers include:

3.4.1 RDF Bag

A Bag (a resource having type `rdf:Bag`) represents a group of property values where there is no significance in the order of the members. A Bag might be used to describe a group of authors in which the order of entry or processing does not matter.

3.4.2 RDF Sequence

A Sequence or Seq (a resource having type `rdf:Seq`) represents a group of property values where the order of the members is significant. For example, a Sequence might be used to describe a group that must be maintained in alphabetical order.

3.4.3 RDF Alternative

An Alternative or Alt (a resource having type `rdf:Alt`) represents a group of property values that are *alternatives* (typically for a single value of a property). For example, an Alt might be used to describe alternative names for an author.

The members of the container can be described by defining a **container membership property** for each member.. These container membership properties may have names of the form `rdf:_`*n*, where *n* is a decimal integer greater than zero, with no leading zeros, e.g., `rdf:_1`, `rdf:_2`, `rdf:_3`, and so on, and are used specifically for describing the members of containers. Or the container membership properties may have names of the form `rdf:li` (list item) for the convenience of not having to explicitly number each membership property

Grouped Property Values are not used in any examples within this document. Note, however, that these RDF structures may be used with metadata fields defined for the `prism:` namespace. See Example 6.0.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:prism="http://prismstandard.org/namespaces/1.2/basic/">
   <rdf:Description rdf:about="http://example.org/courses/6.001">
      <prism:category>
         <rdf:Bag>
            <rdf:li> category.xml#newsBulletin</rdf:li>
            <rdf:li> category.xml#electionResults </rdf:li>
         </rdf:Bag>
      </prism:category>
   </rdf:Description>
</rdf:RDF>
```

Example 6.0 RDF Container Elements

4 PRISM XML/RDF Element and Attribute Definitions

XML/RDF content and attribute models are defined with keywords in Table 3.0 for use in documenting the XML/RDF Element and Attributes within PRISM.

Representation	Definition
URI Resource	This specifies that the property element (that is, the element specifying a particular property of the subject) is EMPTY and that the value is specified using a URI Resource attribute value.
Authority Reference	This specifies that the property value is specified using a kind of URI Reference where the attribute, “rdf:resource”, has a value that is a URI referring to a term in a controlled vocabulary.
Resource Reference	This specifies the requirement of the attribute, “rdf:resource”, whose value is a URI reference to a resource. The set of AuthorityReferences is a subset of the set of ResourceReferences.
Plain Literal	This specifies that a plain literal will be used to provide the property value within an element.
Enumerated Literal	This specifies that a plain literal with specifically enumerated values will be used to provide the property value within an element. Note that RDF does not support the concept of an enumerated literal, but XSD, RNG, and DTD attribute specifications do.
XML Literal	This specifies that an XML literal content model will be used to specify the property value within an element. In this case, the rdf:parseType must be specified as “Literal”
Typed Literal	This specifies that a typed literal is being use to specify the property value within an element. The attribute rdf:datatype must be specified to indicate the datatype of the element content
Resource Node	This specifies that the property element contains other property element nodes. The attribute rdf:parseType must be specified to be “Resource”

Table 3.0 Keywords for XML/RDF Element and Attribute Definitions

4.1 PRISM Rights Language Namespace

The PRISM WG put only the most commonly-needed rights elements into the PRISM namespace. For more involved treatment of rights and permissions in PRISM descriptions, elements from another namespace must be used. Because of the considerable activity around specifying rights and permissions, the PRISM Working Group could not recommend an existing standard to follow, as they were able to do with XML, RDF, and the Dublin Core. Therefore the working group has defined a small, simple, extensible language for expressing common rights and permissions. That language is known as the PRISM Rights Language (PRL). This section specifies that language. Note that implementations of PRISM MAY also implement PRL, but it is not mandatory. The PRISM Working Group expects PRL to be supplanted in time, once the activity around many different rights languages has settled down.

4.2 Processing Model

Collections of PRL statements are known as PRL expressions. The purpose of a PRL expression is to determine if a person or organization may or may not make use of a resource in a particular way. PRL expressions evaluate to a Boolean value that indicates if a particular use is allowed (if the expression evaluates to true) or not (if the expression evaluates to false).

PRL evaluation is described in RDF domain, not in the XML syntax domain. Note that PRL expressions do not describe the resource directly. They describe the real or virtual agreement under which the sender and receiver are operating. PRL expressions consist of one or more clauses. A clause, in the RDF domain, is a resource that represents a real or virtual clause in the agreement between the sender and receiver. It is the RDF subject of statements that convey the intent of the clause. In PRISM descriptions, PRL expressions MUST appear only

within the scope of a dc:rights element. The dc:rights statement contains the clause, or an rdf:Bag element if there are multiple clauses.

Each clause has a possibly empty set of usage statements and a possibly empty set of condition statements. If no usage is specified, the default usage is #use. (#use will be defined later in this section). If no conditions are specified, the default condition evaluates to 'true'.

Conditions evaluate to Boolean true or false. Conditions are expressed in XML using elements from the PRL namespace, such as prl:geographic and prl:industry. Two elements from the PRISM namespace, prism:embargoDate and prism:expirationDate, also express PRL conditions. To evaluate a condition, a comparison is made between the value(s) supplied in the XML element and the current state of the system or the intended use of content. The exact nature of the comparison depends on the condition being tested. True values mean that the condition applies. For example, the prism:embargoDate condition evaluates to 'true' if the current system date and time is greater than or equal to the date and time specified in that element's content. The prl:industry condition evaluates to 'true' if the content is intended to be used in the specified industry. This specification does not define how the current state of the system and the intended use(s) of the content are made available for evaluating the conditions.

Usages do not evaluate to Booleans. Instead, they evaluate to a set of URI references (which is typically of length 1). The URI references govern what the receiving system can do with the described resource. PRL defines only the four URI references shown in [PRISMCOMP], Rights and Usage Vocabularies. Others can be defined, but this is expected to be an exceedingly rare form of extension.

To evaluate a clause, the logical AND of the conditions in the clause is computed. If that is false, the clause evaluates to the PRL usage #notApplicable. If the logical AND is true, the set of usages in the clause is evaluated and returned as the value of the clause.

To evaluate a PRL expression, all the clauses are evaluated and their results are merged according to the following rules, which MUST be applied in the following order:

1. U, the UNION of the sets of URI references is computed. If multiple PRL expressions exist because the described resource had multiple dc:rights elements, those usages are also included in the computation of U.
2. If #none is a member of U, the expression evaluates to false.
3. Any special rules needed by extension elements are applied.
4. If #use is a member of U, the expression evaluates to true .

If the PRL expression evaluates to true, the resource may be used. If it evaluates to false, it may not be used. Typically, human intervention at runtime will be needed to convert the URI references, such as #permissionsUnkown, to a Boolean value.

Note that because PRL defines both #none and #use, the NOT operator is not needed.

PRL can be extended by defining new conditions and usages in other namespaces. Conditions MUST be defined to return a Boolean where true means the condition applies to the current state of the system or intended use of the content. Also, the conditions MUST be side-effect-free. Usages MUST return a URI reference. Another extension mechanism exists in PRL. The content model of the prl:usage element allows text content. When text content is given, implementations MUST convert it to a URI reference. This specification does not specify how that is to happen, however, a common means of doing so is expected to be showing the text to a user and asking them if the result should be #use or #none.

4.3 PRISM XML/RDF Element and Attribute Models

In combining XML with RDF, there is far greater flexibility in tagging than we are used to when we define XML elements and attributes with an XML DTD. The remainder of this section contains the most likely element/attribute models for profile 2 PRISM. Other profile 2 models are possible based on the interaction between XML and RDF.

4.3.1 prl:geography

Name	Geography (as condition on use of a resource)
Identifier	prl:geography
Definition	Name of, or authority file reference to, a geographic region of interest.
Comment	Recommended practice is to use the ISO 3166-1 and 3166-2 country and region codes. For profile 2, if there is more than one geography is related to a resource, PRISM recommends listing the multiple locations inside one prl:geography element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile 1, just repeat the prl:geography element multiple times.
Model #1	
Element Content	URI Resource (no element content)
Attributes	Authority Reference.(rdf:resource)
Model #2	
Element Content	Plain Literal
Attributes	xml:lang (optional) designed for identifying the human language used
Model #3	
Element Content	XML Literal
Attributes	rdf:parseType=”Literal” xml:lang (optional) designed for identifying the human language used
Occurs In	PRL clauses, which are contained in or referred to by a dc:rights element.
Occurance	May occur 0 or many times
Example	Model #1 <prl:geography rdf:resource= ”http://prismstandard.org/vocabs/ISO-3166/GB”/> Model #2 <prl:geography>Oklahoma</prl:geography> Model #3 <prl:geography rdf:parseType=”Literal”> South & East Counties </prl>

4.3.2 prl:industry

Name	Industry (as condition on use of a resource)
Identifier	prl:industry
Definition	Name of, or authority file reference to, an industry or industrial sector of interest.
Comment	Recommended practice is to specify the industry sector using the NAICS industrial classification system. For profile 2, if there is more than one industry related to a resource, PRISM recommends listing the multiple industries inside one prl:industry element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile 1, just repeat the prl:industry element multiple times.
Model #1	
Element Content	URI Resource (no element content)
Attributes	Authority Reference.(rdf:resource)
Model #2	
Element Content	Plain Literal
Attributes	xml:lang (optional) designed for identifying the human language used
Model #3	
Element Content	XML Literal
Attributes	rdf:parseType=”Literal” xml:lang (optional) designed for identifying the human language used
Occurs In	PRL clauses, which are contained in or referred to by a dc:rights element.
Occurence	May occur 0 or many times
Example	Model #1 <prl:industry rdf:resource= ” http://www.census.gov/epcd/naics02/naicod02.htm#11111”/> Model #2 <prl:industry> 11111</prl:industry> Model #3 <prl:industry rdf:parseType=”Literal”> Postharvest Crop Activities (except cotton ginning) </prl>

4.3.3 prl:usage

Name	Resource Usage
Identifier	prism:usage
Definition	Authority reference or human-readable description of a use that is allowed or restricted. Authority references SHOULD reference values from [PRISMCOMP] Table 4 Predefined Usages.
Comment	For profile 2, if there is more than one usage statement is related to a resource, PRISM recommends listing the multiple usage locations inside one prl:usage element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile 1, just repeat the prl:usage element multiple times.
Model #1	
Element Content	URI Resource (no element content)
Attributes	Authority Reference.(rdf:resource)
Model #2	
Element Content	Plain Literal
Attributes	xml:lang (optional) designed for identifying the human language used
Model #3	
Element Content	XML Literal
Attributes	rdf:parseType=”Literal” xml:lang (optional) designed for identifying the human language used
Occurs In	PRL clauses, which are contained in or referred to by a dc:rights element.
Example	Model #1 <prl:usage rdf:resource= ” http://www.prismstandard.org/vocabularies1.2/usage.xml#notReusable”/> Model #2 <prl:usage> May not use on keychains or coffee mugs</prl:usage> Model #3 <prl:usage><para>May not use on <item>keychains</item> or <item>coffee mugs</item></para></prl:usage>

Bibliography for the PRISM 1.3 Documentation Package

Normative References

[AAT] Getty Art and Architecture Thesaurus. <http://shiva.pub.getty.edu/aat_browser/>

[DCMI] Dublin Core Metadata Element Set, Version 1.1: Reference Description. http://purl.org/dc/documents/rec-dces-19990702.htm

[DCMES-XML] Expressing Simple Dublin Core in RDF/XML, Dave Becket, Eric Miller and Dan Brickley Editors. Recommendation 2002-07-31, http://dublincore.org/documents/dcmes-xml/

[DCMI-TERMS] Dublin Core Metadata Terms, 2005-01-10; .
http://dublincore.org/documents/2005/01/10/dcmi-terms/

[IETF-MIMETYPES] Internet Assigned Numbers Authority (IANA); Internet Media Types. http://www.isi.edu/in-notes/iana/assignments/media-types/media-types

[IETF-XML-Media] M. Murata, S. St.Laurent, D. Kohn; XML Media Types; Jan. 2001. http://www.ietf.org/rfc/rfc3023.txt

[IETF RFC 3066], Tags for the Identification of Languages ed. H. Alvestrand. 2001. http://www.ietf.org/rfc/rfc3066.txt.

[IPTC-NEWSML] International Press and Telecommunications Council, NewsML Specification & Documents; http://www.iptc.org/site/NewsML/NewsMLSpec.htm

[IPTC-NITF] International Press and Telecommunications Council, News Industry Text Format.

http://www.nitf.org/html/tech-nitf.html

[ISO-639] ISO 639 - Codes for the representation of names of languages. http://www.oasis-open.org/cover/iso639a.html

[ISO-3166] ISO 3166 - Codes for the representation of names of countries and their subdivisions. http://www.din.de/gremien/nas/nabd/iso3166ma/a3ptnorm.html

[NAICS] North American Industry Classification System; 1997. http://www.census.gov/epcd/www/naics.html

[RFC-3066] H. Alvestrand; Tags for the Identification of Languages; January 2001. http://www.ietf.org/rfc/rfc3066.txt

[IETF-MediaTypes] N. Freed & N. Borenstein. November 1996, Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types. http://www.ietf.org/rfc/rfc2046.txt

[PRISMINT] PRISM Working Group, 2005, PRISM Introduction, v 1.3, www.prismstandard.org

[PRISMPRISMNS] PRISM Working Group, 2005, The PRISM Namespace v 1.3, www.prismstandard.org

[PRISMCOMP] PRISM Working Group, 2005, PRISM Compliance, v 1.3, www.prismstandard.org

[PRISMDCNS] PRISM Working Group, 2005, The PRISM Subset of the Dublin Core Namespace v 1.3, www.prismstandard.org

[PRISMRLNS] PRISM Working Group, 2005, The PRISM Rights Language Namespace v 1.3, www.prismstandard.org

[PRISMCVNS] PRISM Working Group, 2005, The PRISM Controlled Vocabulary Namespace v 1.3, www.prismstandard.org

[PRISMIMNS] PRISM Working Group, 2005, The PRISM Inline Markup Namespace v 1.3, www.prismstandard.org

[PRISMAMNS] PRISM Working Group, 2005, The PRISM Aggregator Message Namespace v 1.2, www.prismstandard.org

[PAMGUIDE] PRISM Working Group, 2004, Guide to the PRISM Aggregator Document Type Definition (DTD) V. 1.1, www.prismstandard.org

[RFC-2119] S. Bradner, Key words for use in RFCs to Indicate Requirement Level http://www.ietf.org/rfc/rfc2119.txt

[RFC-2396] Uniform Resource Identifiers (URI): Generic Syntax, Internet RFC 2396. http://www.ietf.org/rfc/rfc2396.txt

[TGN] Getty Thesaurus of Geographic Names. http://shiva.pub.getty.edu/tgn_browser/

[W3C-DateTime] Misha Wolf, Charles Wicksteed; Date and Time Formats; W3C Note; http://www.w3.org/TR/NOTE-datetime.html

[W3C-RDF] Ora Lassila, Ralph R Swick, Resource Definition Framework (RDF) Model and Syntax Specification. http://www.w3.org/TR/REC-rdf-syntax

[W3C-RDFS] Dan Brickley, R.V. Guha (eds.), Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation, 27 March 2000,http://www.w3.org/TR/2000/CR-rdf-schema-20000327

[W3C-XML] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen (eds.), Extensible Markup Language (XML) http://www.w3.org/TR/REC-xml

[W3C-XML-BASE] Jonathan Marsh (ed.); XML Base; http://www.w3.org/TR/xmlbase/

[W3C-XML-NS] Tim Bray, Dave Hollander, Andrew Layman (eds.); Namespaces in XML. http://www.w3.org/TR/REC-xml-names

[XMP] XMP Specification, Adobe Systems. http://partners.adobe.com/public/developer/en/xmp/sdk/xmpspecification.pdf

Non-Normative References

[DCMI-R] Relation Element Working Draft; Dublin Core Metadata Initiative; 1997-12-19.

[ICE] The Information and Content Exchange (ICE) Protocol. http://www.w3.org/TR/NOTE-ice.html , http://www.icestandard.org, http://www.icestandard.org/Spec/SPECICE-2.0.pdf

[ISO-8601] ISO (International Organization for Standardization), ISO 8601:1988 (E) Data elements and interchange formats - Information interchange - Representation of dates and times, 1998. http://www.iso.ch/cate/d15903.html

[ISO-13250] ISO/IEC 13250 Topic Maps: Information Technology -- Document Description and Markup Languages.

[ODRL] Open Digital Rights Language, Version 1.0; Dec. 2001. http://odrl.net

[RDDL] ….

[RDF-EASY] *RDF Made Easy*, by John Cowan: http://home.ccil.org/~cowan/XML/RDF-made-easy.html

[RDF-FRIENDLY] *Making Your XML RDF-Friendly* by John Cowan and Bob DuCharme: http://www.xml.com/pub/a/2002/10/30/rdf-friendly.html

[RDF-MIME-TYPE] *MIME Media Types*, The Internet Assigned Numbers Authority (IANA). This document is http://www.iana.org/assignments/media-types/ . The registration for `application/rdf+xml` is archived at http://www.w3.org/2001/sw/RDFCore/mediatype-registration .

[RDF-PRIMER] *RDF Primer*, Frank Manola and Eric Miller, Editors, W3C Recommendation, 10 February 2004, http://www.w3.org/TR/2004/REC-rdf-primer-20040210/ . Latest version available at http://www.w3.org/TR/rdf-primer/ .

[RDF-SEMANTICS] *RDF Semantics*, Patrick Hayes, Editor, W3C Recommendation, 10 February 2004, http://www.w3.org/TR/2004/REC-rdf-mt-20040210/ . Latest version available at http://www.w3.org/TR/rdf-mt/ .

[RDF-SYNTAX] *RDF/XML Syntax Specification (Revised)*, Dave Beckett, Editor, W3C Recommendation, 10 February 2004, http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/ . Latest version available at http://www.w3.org/TR/rdf-syntax-grammar/ .

[RDF-VOCABULARY] *RDF Vocabulary Description Language 1.0: RDF Schema*, Dan Brickley and R. V. Guha, Editors, W3C Recommendation, 10 February 2004, http://www.w3.org/TR/2004/REC-rdf-schema-20040210/ . Latest version available at http://www.w3.org/TR/rdf-schema/ .

[TZ-LIB] Time Zone Library; ftp://elsie.nci.nih.gov/pub/

[W3C-SMIL] Synchronized Multimedia Integration Language (SMIL) 1.0 Specification (SMIL) http://www.w3.org/TR/Rec-SMIL

[XrML] ContentGuard, Inc., Extensible Rights Markup Language. http://www.xrml.org/

[XML-1.1] *Extensible Markup Language (XML) 1.1*, John Cowan, Editor. W3C Candidate Recommendation 15 October 2002. This version is http://www.w3.org/TR/2002/CR-xml11-20021015/. The latest version is available at http://www.w3.org/TR/xml11/.

[XML-SCHEMA1] *XML Schema Part 1: Structures* W3C Recommendation, World Wide Web Consortium, 2 May 2001. This version is http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/. The latest version is available at http://www.w3.org/TR/xmlschema-1/.

[XML-SCHEMA2] *XML Schema Part 2: Datatypes*, W3C Recommendation, World Wide Web Consortium, 2 May 2001.This version is http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/. The latest version is available at http://www.w3.org/TR/xmlschema-2/.