

PRISM:
Publishing Requirements for Industry Standard Metadata
PRISM Specification: Modular: Version 1.3

PRISM Subset of the Dublin Core Namespace

2005 10 01

Copyright and Legal Notices

Copyright (c) International Digital Enterprise Alliance, Inc. [IDEAlliance] (2001, 2001, 2003, 2004, 2005).

All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to IDEAlliance, except as needed for the purpose of developing IDEAlliance specifications, in which case the procedures for copyrights defined in the IDEAlliance Intellectual Property Policy document must be followed, or as required to translate it into languages other than English. The limited permissions granted above are perpetual and will not be revoked by IDEAlliance or its successors or assigns.

NO WARRANTY, EXPRESSED OR IMPLIED, IS MADE REGARDING THE ACCURACY, ADEQUACY, COMPLETENESS, LEGALITY, RELIABILITY OR USEFULNESS OF ANY INFORMATION CONTAINED IN THIS DOCUMENT OR IN ANY SPECIFICATION OR OTHER PRODUCT OR SERVICE PRODUCED OR SPONSORED BY IDEALLIANCE. THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN AND INCLUDED IN ANY SPECIFICATION OR OTHER PRODUCT OR SERVICE OF IDEALLIANCE IS PROVIDED ON AN "AS IS" BASIS. IDEALLIANCE DISCLAIMS ALL WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, ANY ACTUAL OR ASSERTED WARRANTY OF NON-INFRINGEMENT OF PROPRIETARY RIGHTS, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.

NEITHER IDEALLIANCE NOR ITS CONTRIBUTORS SHALL BE HELD LIABLE FOR ANY IMPROPER OR INCORRECT USE OF INFORMATION. NEITHER IDEALLIANCE NOR ITS CONTRIBUTORS ASSUME ANY RESPONSIBILITY FOR ANYONE'S USE OF INFORMATION PROVIDED BY IDEALLIANCE. IN NO EVENT SHALL IDEALLIANCE OR ITS CONTRIBUTORS BE LIABLE TO ANYONE FOR DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO, COMPENSATORY DAMAGES, LOST PROFITS, LOST DATA OR ANY FORM OF SPECIAL, INCIDENTAL, INDIRECT, CONSEQUENTIAL OR PUNITIVE DAMAGES OF ANY KIND WHETHER BASED ON BREACH OF CONTRACT OR WARRANTY, TORT, PRODUCT LIABILITY OR OTHERWISE.

IDEAlliance takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available. IDEAlliance does not represent that it has made any effort to identify any such rights. Information on IDEAlliance's procedures with respect to rights in IDEAlliance specifications can be found at the IDEAlliance website. Copies of claims of rights made available for publication, assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification, can be obtained from the President of IDEAlliance.

IDEAlliance requests interested parties to disclose any copyrights, trademarks, service marks, patents, patent applications, or other proprietary or intellectual property rights which may cover technology that may be required to implement this specification. Please address the information to the President of IDEAlliance.

Table of Contents

1	Status	1
1.1	Document Status	1
1.2	Document Location	1
1.3	Version History	1
2	PRISM Documentation Structure	3
2.1	Normative and Non-normative Sections	3
2.1.1	Requirement Wording Note	3
2.2	The PRISM Documentation Package	3
2.2.1	Additional PRISM Documentation	4
2.2.2	Access to PRISM Documentation.....	4
3	Introduction.....	5
3.1	Purpose and Scope.....	5
3.2	About RDF.....	5
3.3	Specifying Nodes	6
3.3.1	URI References	6
3.3.2	Literals.....	6
3.3.2.1	Plain Literals.....	6
3.3.2.2	Typed Literal.....	6
3.3.2.3	XML Literals	7
3.3.3	Nodes Made of Elements.....	7
3.4	Grouped Property Values.....	8
3.4.1	RDF Bag	8
3.4.2	RDF Sequence	8
3.4.3	RDF Alternative.....	8
4	PRISM XML/RDF Element and Attribute Definitions.....	10
4.1	Dublin Core Namespace	10
4.2	PRISM Subset of Dublin Core XML/RDF Element and Attribute Models	10
4.2.1	dc:contributor	11
4.2.2	dc:coverage	12
4.2.3	dc:creator	13
4.2.4	dc:date	14
4.2.5	dc:description	15
4.2.6	dc:format	16

4.2.7	dcterms:hasFormat	17
4.2.8	dcterms:hasPart	18
4.2.9	dcterms:hasVersion	19
4.2.10	dc:identifier	20
4.2.11	dcterms:isPartOf	21
4.2.12	dcterms:isReferencedBy	22
	dcterms:isRequiredBy	23
4.2.13	dcterms:isVersionOf	24
4.2.14	dc:language	25
4.2.15	dc:publisher	25
4.2.16	dc:relation	26
4.2.17	dc:rights	26
4.2.18	dc:source	27
4.2.19	dc:subject	28
4.2.20	dc:title	29
4.2.21	dc:type	30
	Bibliography for the PRISM 1.3 Documentation Package	31

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_dublin_core_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
1.3 Draft A	5/30/05	Kennedy/McConnell	Enhance element descriptions and examples. Include RDF discussion as per edits to [PRISM/PRISMNS]
1.3 Draft B	6/6/05	Kennedy	Resolve comments from WG Con Calls
1.3 Final	10/01/05	Kennedy	Resolve open 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:

<i>Document</i>	<i>Description</i>
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 elements that PRISM includes from the Dublin Core namespace. For the Dublin Core specification, see [DCMI-TERMS]. All of section 4 of this document is normative.

All the element definitions appear in a uniform format. Each element definition begins with two fields – the Name and the Identifier of the element. The Name is a human-readable string that can be translated into different languages. Also, note that PRISM does NOT require that users be presented with the same labels. The Identifier is a protocol element. It is an XML element type and MUST be given as shown, modulo the normal allowance for variations in the namespace prefix used.

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.

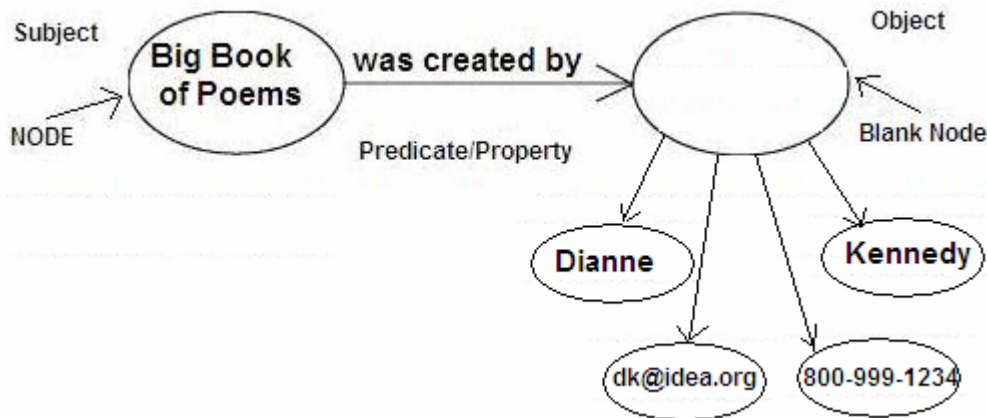


Figure 3.0 RDF Graph with a Blank Node

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

While a blank node can occur anywhere within PRISM metadata fields, some elements from the PRISM subset of Dublin Core are more likely to be modeled as blank nodes than others. These elements are listed in Table 2.0.

Name	Identifier	Comment
Description	dc:description	The description may be modeled as being made up of a number of other elements from other namespaces. For example, one might model dc:description using dc:abstract and dc:educationLevel.
Rights	dc:rights	Dublin Core Rights may be modeled as being made up of a number of other elements from other name spaces. For example, one might model dc:rights using elements from the prism: or prl namespace.

Table 2.0 Possible Blank Nodes from prism: Namespace

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 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 `dc:` namespace. See Example 6.0.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description rdf:about="http://example.org/courses/6.001">
    <dc:creator>
      <rdf:Bag>
        <rdf:li>Linda Burman</rdf:li>
        <rdf:li>Ron Daniel</rdf:li>
      </rdf:Bag>
    </dc:creator>
  </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 Authority References is a subset of the set of Resource References.
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 used 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 Dublin Core Namespace

The normative definitions of the Dublin Core elements can be found in [DCMI] [Dublin Core Metadata Element Set, Version 1.1](#) and [DCMI-TERMS] [Dublin Core Metadata Terms, 2005-01-10](#). The use of some dc: elements is encouraged, others are discouraged, and others constrained. New dcterms: elements have been added in version 1.3.

None of the Dublin Core elements are required to appear in a PRISM description -- except dc:identifier, under profile one compliance; see [PRISMCOMP] -- and all of them are repeatable any number of times.

4.2 PRISM Subset of Dublin Core 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. Other models are possible based on the interaction between XML and RDF.

4.2.1 dc:contributor

Name	Contributor
Identifier	dc:contributor
Definition	An entity responsible for making contributions to the content of the resource.
Comment	<p>Dublin core recommends that dc:contributor identifies a person, an organization, or a service by name.</p> <p>PRISM recommends that magazine publishers use dc:contributor for people who do additional reporting, or individuals who would be called out for special acknowledgments, such as research assistants. Individuals who would be credited for hair, makeup, etc. would typically NOT be listed in dc:contributor. Instead, such credits are expected to be provided in the marked-up article, but not in the metadata for the article. Individuals called out for special acknowledgements, such as research assistants, would be listed in dc:contributor elements.</p> <p>For profile 2, If there are multiple contributors for the resource PRISM recommends listing the multiple contributors inside one dc:contributor element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile 1, simply repeat the dc:contributor element multiple times.</p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Authority Reference (rdf:resource)
Model #2	
Element Content	Plain Literal
Attributes	xml:lang (optional) designed for identifying the human language used.
Occurs in	
Occurrence	May occur 0 to many times
Example	<p>Model #1 <code><dc:contributor rdf:resource="http://wanderlust.com/jas"/></code></p> <p>Model #2 <code><dc:contributor>John Smith</dc:contributor></code></p>

4.2.2 dc:coverage

Name	Coverage
Identifier	dc:coverage
Definition	The spatial and/or temporal extent of the content of the resource.
Comment	Dublin core recommends that dc:coverage will typically include spatial location (a place name or geographic coordinates), temporal period (a period label, date, or date range). PRISM recommends use of dc:coverage only for temporal subjects of the resource. PRISM's recommended best practice is to use prism:location for cases where a geographic area is a subject of the resource.
Model #1	
Element Content	URI Reference (empty element)
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
Occurrence	May occur 0 or 1 time
Example	Model #1 <dc:coverage rdf:resource= http://www.timeframe.com/Mauve Decade/ > Model #2 <dc:coverage>ca. 1200 B.C.</dc:coverage> Model #3 <dc:coverage>1968&emdsh;1972</dc:coverage>

4.2.3 dc:creator

Name	Creator
Identifier	dc:creator
Definition	An entity primarily responsible for making the content of the resource.
Comment	<p>Dublin core recommends that dc:creator includes a person, an organization, or a service. The Creator may be identified by name, or by reference to an entry in a controlled vocabulary.</p> <p>PRISM's recommendation for magazine publishing is for dc:creator to contain the same as the byline (in most cases this would be the writer or writers).</p> <p>PRISM's recommendation for magazine publishing is for dc:creator to contain the same as the byline (in most cases this would be the writer or writers).</p> <p>In principle, any number of creators may be associated with a resource. PRISM recommends that this element contain the name of one person or organization primarily responsible for the intellectual content of the resource.</p> <p>For profile 2, If there are multiple contributors for the resource PRISM recommends listing the multiple creators inside one dc:creator element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile 1, simple repeat the dc:creator element multiple times.</p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 0 or more times
Example	<pre> Model #1 <dc:creator rdf:resource=http://www.timeinc.com/pmeier/> Model #2 <dc:creator> <rdf:Bag> <rdf:li>Anne Considine </rdf:li> <rdf:li>Tina Steil</rdf:li> <rdf:li>Lee Vetten</rdf:li> </rdf:Bag> </dc:creator> Model #3 <dc:creator>Williams & Sonoma</dc:creator> </pre>

4.2.4 dc:date

Name	Date
Identifier	dc:date
Definition	A date associated with an event in the life cycle of the resource.
Comment	<p>Dublin core defines dc:date as any date associated with the creation or availability of the resource. The Dublin Core definition of date is quite loose; therefore, PRISM recommends that this element not be used, other than in the exceptional cases mentioned below. If it is used, its meaning SHOULD be used for the cover date of the magazine in which the resource appeared. One case in which PRISM recommends the use of this element is when the publication date is not specific to a day, month, or year.</p> <p>For example, "Spring, 2002" should go into prism:coverDisplayDate. In such cases the non-specific publication date should be provided in a dc:date element, and a more specific publication date (if available) should be provided in the prism:publicationDate element.</p>
Element Content	Typed Literal
Attributes	rdf:datatype must be specified as "http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/#dateTime" must also be specified. Typed literal is a string specifying a date and time according to the W3C profile of ISO 8601 (e.g., YYYY-MM-DDThh:mm:ss.ssTZD) Note that this includes time zone data which may be important (see PRISM:publicationDate)
Occurrence	May occur 0 or 1 time
Example	<code><dc:date rdf:datatype="http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/#dateTime">2001-04-09</dc:date></code>

4.2.5 dc:description

Name	Description
Identifier	dc:description
Definition	An account of the content of the resource.
Comment	<p>The Dublin Core Metadata Initiative recommends that dc:description MAY contain any information (e.g., an abstract, table of contents, reference to a graphical representation of content or a free-text account of the content) that describes the resource.</p> <p>For PRISM descriptions, the element provides material that describes the resource, such as an abstract or a deck head. Note that this is intended to appear in metadata for an article, not as inline markup. (In other words, a DTD for articles might have dc:description in the header, but would use elements like <abstract> or <deck> for the markup of such material in the body of the article).</p> <p>Short descriptions, such as those which appear in the Table of Contents of a magazine, or might appear in the results list of an online search, SHOULD be given in the prism:teaser element.</p> <p>The dc:description element MAY refer to separate descriptions, such as an abstract prepared by an A&I service, by providing the URI of the description as the value of an rdf:resource attribute. (In this case, the description is a separate, standalone resource which could have its own metadata. The metadata record for the separate abstract should contain a <prism:category> of abstract, and a <dc:source> element pointing back to the original article.) For XMP compatibility, we recommend coding multiple descriptions using RDF grouping elements such as <rdf:Bag>.</p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 0 or more times
Example	<pre> Model #1 <dc:description rdf:resource= "http://www2.rhbnc.ac.uk/Music/Archive/Disserts/attinell.html"/> Model #2 <dc:description>Browse our catalog of desktop and notebook computers to find one just right for you.</dc:description> Model #3 <dc:description rdf:parseType="Literal"> Describes the infamous criminal and gunfighter, Billy the Kid. </dc:description> </pre>

4.2.6 dc:format

Name	Format
Identifier	dc:format
Definition	The physical or digital manifestation of the resource.
Comment	<p>Dublin core recommends that dc:format may include the media-type or dimensions of the resource. Format may be used to determine the software, hardware or other equipment needed to display or operate the resource. Examples of dimensions include size and duration.</p> <p>PRISM focuses on systems where resources are digital content, not physical objects. Therefore, PRISM-compliant systems sending PRISM records MUST restrict values of the dc:format element to those in list of Internet Media Types [MIME]. Since the Dublin Core specification does not impose that restriction, PRISM-compliant systems receiving descriptions MAY wish to detect when format values are strings other than media types in order to allow application-appropriate handling.</p>
Element Content	Plain Literal
Attributes	xml:lang (optional) designed for identifying the human language used.
Occurrence	May occur 0 or 1 time
Example	<dc:format>application/pdf</dc:format>

4.2.7 dcterms:hasFormat

Name	Has Format
Identifier	prism:hasFormat
Definition	Identifies another resource, which is essentially the same intellectual content as the current resource, but presented in another file format, or after some mechanical transformation like a different resolution, different color depth, etc.
Comment	<p>The dcterms:hasFormat element points from the original resource, to the alternative version derived from it. In other words, the metadata of the original resource will contain the prism:hasFormat element. The dcterms:isFormatOf element is used to point in the other direction, from the alternative back to the original. If the 'original' version cannot be determined, use prism:hasFormat for both directions of the relationship.</p> <p>The element may be repeated if there are multiple alternative formats for the resource. For XMP compatibility, we recommend using RDF grouping elements such as <rdf:Bag>.</p> <p>Note: <i>This new Dublin Core element in the PRISM Subset is equivalent to the prism:hasFormat. The prism:hasFormat will be deprecated over time and replaced by dcterms:hasFormat.</i></p>
Model #1	
Element Content	URI Reference
Attributes	Resource 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
Occurrence	May occur 0 or more times
Example	<pre> Model #1 <dcterms:hasFormat rdf:resource="doi:123/p92-1293" /> Model #2 <dcterms:hasFormat>photo1293.jpg</dc:hasFormat> Model #3 <dcterms:hasFormat rdf:parseType="Literal"> doi&colon:123/p92&endash;1293</dcterms:hasFormat> </pre>

4.2.8 dcterms:hasPart

Name	Has Part
Identifier	dc:hasPart
Definition	The described resource includes the referenced resource either physically or logically.
Comment	<p>dc:hasPart allows the metadata for an article to identify images, sidebars, tables, graphs, maps, illustrations, etc. in the article which exist as separate, identifiable, resources. The metadata for those resources can then be fetched, based on the identifier for the included resource.</p> <p>Recommended best practice is to describe photos, etc. as separate objects, rather than embedding their metadata in the metadata for an article, in order to ease their reuse and to simplify data maintenance when the resources are reused. Best practice is also to identify the resources with URIs, rather than human-readable text descriptions, in order to enable automated handling of the resource.</p> <p>If there are multiple parts for the resource, PRISM recommends listing the multiple formats inside one dcterms:hasPart element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible.</p> <p>Note: This new Dublin Core element in the PRISM Subset is equivalent to the prism:hasPart. The prism:hasPart will be deprecated over time and replaced by dcterms:hasPart.</p>
Model #1	
Element Content	URI Reference
Attributes	Resource 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
Occurrence	May occur 0 or more times.
Example	<p>Model #1 <dcterms:hasPart rdf:resource="http://www.travelmongo.com/2000/08/BelizePhoto.jpg" /></p> <p>Model #2 <dcterms:hasPart>http://www.travelmongo.com/2000/08/BelizePhoto.jpg</dcterms:hasPart></p> <p>Model #3 <dcterms:hasPart rdf:parseType="Literal">dam&endash;obj&endash;32485u2</dcterms:hasPart></p>

4.2.9 dcterms:hasVersion

Name	Has Version
Identifier	dcterms:hasVersion
Definition	Identifies a version, edition, or adaptation of the current resource.
Comment	<p>Changes in version imply substantive changes in intellectual content rather than differences in format.</p> <p>For changes in format, use the prism:hasFormat element. For the special case of versions known as “corrections”, use prism:hasCorrection to point from the current resource to correction blocks. Use dcterms:hasPreviousVersion to point from the corrected resource back to the earlier one.</p> <p>Note: <i>This new Dublin Core element in the PRISM Subset is equivalent to the prism:hasPreviousVersion. prism:hasPreviousVersion will be deprecated over time and replaced by dcterms:hasVersion.</i></p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 0 or 1 time
Example	<pre> Model #1 <dcterms:hasVersion rdf:resource= "http://travelmongo.com/2000/08/BelizeTravelUpdate.xml" /> Model #2 <dcterms:hasVersion>BelizeTravelUpdate_04.xml </dcterms:hasVersion> Model #3 <dcterms:hasVersion rdf:parseType="Literal"> dam&endash;obj&endash;32485u2</dcterms:hasVersion> </pre>

4.2.10 dc:identifier

Name	Identifier
Identifier	dc:identifier
Definition	An unambiguous reference to the resource, within a given context.
Comment	<p>In PRISM, dc:identifier provides a place for additional identifiers of a resource. In profile two, the rdf:about attribute is always the primary identifier.</p> <p>Recommended best practice is to identify the resource by means of a string or number conforming to a formal identification system. Example formal identification systems include the Uniform Resource Identifier (URI) (including the Uniform Resource Locator (URL)), the Digital Object Identifier (DOI) and the International Standard Book Number (ISBN).</p> <p>Note that multiple dc:identifier statements can be used for internal IDs like, accession number, etc., to identify a particular published item.</p> <p>For profile two, if there are multiple identifiers for the resource, PRISM recommends listing them inside one dc:identifier element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile 1, simple use multiple dc:identifier elements.</p> <p>Consistent and thorough use of identifiers is essential for PRISM conformance.</p> <p>See prism:issueName</p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	0 or more times
Example	<pre> Model #1 <dc:identifier rdf:resource="#chapter1"/> Model #2 <dc:identifier>10-234/3245</dc:identifier> <dc:identifier>Chocolate Desserts issue</dc:identifier> Model #3 <dc:identifier rdf:parseType="Literal">Buyer&apos;s Guide, 2001</dc:identifier> </pre>

4.2.11 dcterms:isPartOf

Name	Is Part Of
Identifier	dcterms:isPartOf
Definition	The described resource is a physical or logical part of the referenced resource.
Comment	<p>This is the inverse of the prism:hasPart relation. Note that it is NOT required to always have both sides of the relationship asserted, as one can be derived from the other.</p> <p>Recommended best practice is to identify the containing resource with a URI. However, textual identifiers are possible so implementations SHOULD be able to accept them, possibly with reduced functionality.</p> <p>Note: <i>This new Dublin Core element in the PRISM Subset is equivalent to the prism:isPartOf. prism:isPartOf will be deprecated over time and replaced by dcterms:isPartOf.</i></p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 0 or 1 time
Example	<p>Model #1 <dcterms:isPartOf rdf:resource="http://TravelMongo.com/2000/08/BelizeArticle.xml" /></p> <p>Model #2 <dctems:isPartOf>BelizeArticle.html</dcterms:isPartOf></p> <p>Model #3 <dcterms:isPartOf rdf:parseType="Literal">dam&endash;obj&endash;32485u2</dcterms:isPartOf></p>

4.2.12 dcterms:isReferencedBy

Name	Is Referenced By
Identifier	dcterms:isReferencedBy
Definition	The described resource is referenced, cited, or otherwise pointed to by the referenced resource. [DCMI- R]
Comment	This is the inverse of the dcterms:references relation. Note: <i>This seems a likely candidate for removal.</i>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Example	<p>Model #1 <code><dcterms:isReferencedBy rdf:resource="http://example.com/documents/dl124352345.xml" /></code></p> <p>Model #2 <code><dcterms:isReferencedBy>dl123452345.xml</dcterms:isReferencedBy></code></p> <p>Model #3 <code><dcterms:isReferencedBy rdf:parseType="Literal">dam&endash;obj&endash;32485u2</dcterms:isReferencedBy></code></p>

dcterms:isRequiredBy

Name	Is Required By
Identifier	dcterms:isRequiredBy
Definition	The described resource is required by the referenced resource, either physically or logically.
Comment	<p>This is the inverse of the dcterms:requires relation.</p> <p>For profile two, PRISM recommends listing the multiple “is required by” statements inside one dcterms:isRequiredBy element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile one, just use multiple dc:isRequiredBy elements.</p> <p>Note: <i>This element was based on a Dublin Core draft specification which has since been approved. The prism:isRequiredBy will be deprecated over time and replaced by dcterms:isRequiredBy.</i></p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 1 or more times
Example	<pre> Model #1 <dcterms:isRequiredBy rdf:resource= "http://wanderlust.com/2000/08/BelizePhoto.jpg" /> Model #2 <dcterms:isRequiredBy>BelizePhoto.jpg</dcterms:isRequiredBy> Model #3 <dcterms:isRequiredBy rdf:parseType="Literal"> dam&endash;obj&endash;32485u2 </dcterms:isRequiredBy> </pre>

4.2.13 dcterms:isVersionOf

Name	Is Version Of
Identifier	dcterms:isVersionOf
Definition	The described resource is a version, edition, or adaptation of the referenced resource. Changes in version imply substantive changes in intellectual content rather than differences in format.
Comment	For corrections, use prism:isCorrectionOf. For alternative versions that do not have substantive changes in intellectual content, use prism:isAlternativeFor. Note: <i>This element was based on a Dublin Core draft specification which has since been approved. The prism:isVersionOf will be deprecated over time and replaced by dcterms:isVersionOf.</i>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 0 or 1 time
Example	<p>Model #1 <code><dcterms:isVersionOf rdf:resource="http://travelmongo.com/2000/08/BelizeTravel.xml" /></code></p> <p>Model #2 <code><dcterms:isVersionOf>http://travelmongo.com/2000/08/BelizeTravel.xml</dcterms:isVersionOf></code></p> <p>Model #3 <code><dcterms:isVersionOf rdf:parseType="Resource">dam&endash;obj&endash;32485u2</dcterms:isVersionOf></code></p>

4.2.14 dc:language

Name	Language
Identifier	dc:language
Definition	A language of the intellectual content of the resource.
Comment	Recommended best practice for the values of the Language element is defined by RFC 3066 [RFC3066]. It specifies the use of a two-letter (or three-letter) Language Code taken from the ISO 639 standard [ISO639] (or from ISO 639-2), optionally followed by a two-letter Country Code (taken from the ISO 3166 standard [ISO3166]). For example, 'en' for English, 'fr' for French, or 'en-GB' for English used in the United Kingdom.
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource Reference (rdf:resource)
Model #2	
Element Content	Plain Literal
Attributes	xml:lang (optional) designed for identifying the human language used
Occurrence	May occur 0 or 1 time
Example	<p>Model #1 <code><dc:language rdf:resource=http://www.din.de/gremien/nas/nabd/iso3166ma/a3ptnorm.htm/></code></p> <p>Model #2 <code><dc:language>en-US</dc:language></code></p>

4.2.15 dc:publisher

Name	Publisher
Identifier	dc:publisher
Definition	An entity responsible for making the resource available.
Comment	The organization or individual that released the resource for publication. For magazine title use PRISM:publicationName.
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 0 or 1 time
Example	<p>Model #1 <code><dc:publisher rdf:resource="http://wanderlust.com/" /></code></p> <p>Model #2 <code><dc:publisher>Wanderlust</dc:publisher></code></p> <p>Model #3 <code><dc:publisher rdf:parseType="literal">Town & Country</dc:publisher></code></p>

4.2.16 dc:relation

Name	Relation
Identifier	dc:relation
Definition	A reference to a related resource.
Comment	Because the notion of “related resource” is vague, PRISM recommends that this element not be used. Preference should be given to the more specific Dublin Core [DCMI-TERMS] and PRISM relationship elements [PRISMPRISM], or to use of the extension mechanisms available in RDF.
Example	No example shown since element is not recommended.

4.2.17 dc:rights

Name	Rights
Identifier	dc:rights
Definition	Information about rights held in and over the resource.
Comment	<p>Typically, a Rights element will contain a rights management statement for the resource, or reference a service providing such information. Rights information often encompasses Intellectual Property Rights (IPR), Copyright, and various Property Rights. If the Rights element is absent, no assumptions can be made about the status of these and other rights with respect to the resource.</p> <p>For PRISM, the dc:rights element specifies the (perhaps implicit) agreement under which the sender allows the receiver to use the content. All rights elements (the PRL elements and the time-specific rights elements) [PRISMLNS] must be contained directly or indirectly in a dc:rights element. Other rights information, such as a copyright statement, that will not vary from one receiver to another may be given as a direct child element of the dc:Description element about the resource.</p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource Reference (rdf:resource)
Model #2	
Element Content	Resource Node
Attributes	Rdf:parseType attribute must be specified as “Resource”
Example	<pre> Model #1 <dc:rights rdf:resource="#standardTerms" /> Model #2 <dc:rights rdf:ParseType="Resource"> <prism:embargoDate>2001-03-01</prism:embargoDate> </dc:rights> </pre>

4.2.18 dc:source

Name	Source
Identifier	dc:source
Definition	A reference to a resource from which the present resource is derived. The present resource is a performance, production, derivation, adaptation or interpretation of the referenced resource.
Comment	This is provided to give appropriate credit to the intellectual heritage of the resource being described when it is an adaptation of another work. When possible, use a URI for an unambiguous reference to the source. Otherwise, a textual identifier of the source may be provided.
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Occurrence	May occur 0 or 1 time
Example	Model #1 <dc:source rdf:resource="http://example.com/classics/Romeo%20and%20Juliet"/> Model #2 <dc:source>Adapted from "The River" by Bruce Springsteen.</dc:source> Model #3 <dc:source>Adapted from The Rive by Bruce Springsteen.</dc:source>

4.2.19 dc:subject

Name	Subject
Identifier	dc:subject
Definition	The topic of the content of the resource.
Comment	<p>Dublin core recommends that dc:subject will be expressed as keywords, key phrases, or classification codes that describe a topic of the resource. Dublin Core and PRISM's recommended best practice is to select a value from a controlled vocabulary, if available.</p> <p>For profile 2, if there are multiple subjects for the resource PRISM recommends listing the multiple subjects inside one dc:subject element using the RDF containers such as rdf:Bag, rdf:Seq or rdf:Alt to be XMP compatible. For profile 1, simply repeat the dc:subject element multiple times.</p> <p>If local operations on the name(s) or definition(s) of the vocabulary elements is needed, PRISM's recommended practice is to provide the value of the dc:subject element using the pcv:Descriptor element and its allowed elements of pcv:vocab, pcv:code, and pcv:label. Remember, PRISM element types are specified in camel case. The exception is that when elements denote Classes in the sense of the RDF Schema [W3C-RDFS], they must begin with an uppercase letter. The only PRISM element to do so is pcv:Descriptor,</p> <p>Note that PRISM defines several elements for more specific types of subjects, such as when events,, locations, organizations, etc. are the subject of the resource. Those elements SHOULD be used in preference to the dc:subject element when they are appropriate.</p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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
Model #4	
Element Content	Resource Node (contains pcv:Descriptor)
Attributes	Rdf:parseType attribute must be specified as "Resource"
Occurs In	For the common case of one company to contact for licensing information, the element SHOULD appear as an immediate child of the rdf:Description element for the resource. In that case it SHALL appear 0 or 1 times. In cases where the rights agent to contact differs from one country to another, or for other reasons, this element SHOULD appear within the scope of a dc:rights element.
Occurrences	May occur 0 or many times
Example	<pre> Model #1 <dc:subject rdf:resource= "http://prismstandard.org/vocabs/lcc/QA76" /> Model #2 <dc:subject>Seasonal Affective Disorder</dc:subject> Model #3 <dc:subject>Seasonal &aff; Disorder</dc:subject> Model #4 <dc:subject rdf:parstType="resource"> <rdf:Bag> <rdf:li>Dogs</rdf:li> <rdf:li>Pets</rdf:li> <rdf:li>Wolves</rdf:li> </rdf:Bag> </dc:Subject> </pre>

4.2.20 dc:title

Name	Title
Identifier	dc:title
Definition	A name given to the resource.
Comment	<p>Dublin core recommends that dc:title will be a name by which the resource is formally known.</p> <p>PRISM recommends that magazine publishers use this for the headline of an article. The name of the magazine in which the article appears can be provided in the prism:publicationName element.</p> <p>The PRISM Specification allows titles to contain special markup characteristics. In such cases the rdf:parseType="Literal" MUST be given.</p>
Model #1	
Element Content	URI Reference (empty element)
Attributes	Resource 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	<p>rdf:parseType="Literal"</p> <p>xml:lang (optional) designed for identifying the human language used</p>
Example	<p>Model #1 <code><dc:title rdf:resource="http://www.usatoday.economy"/></code></p> <p>Model #2 <code><dc:title>Man of the Year, 2002</dc:title></code> <code><prism:publicationName>Time Magazine</prism:publicationName></code> <code><dc:publisher>Time, Inc.</dc:publisher></code></p> <p>Model #3 <code><dc:title rdf:parseType="Literal">E=mc²: The Einstein Myth in 1950's Popular Culture</dc:title></code></p>

4.2.21 dc:type

Name	Type
Identifier	dc:type
Definition	The style of presentation of the resource's content, such as image vs. sidebar.
Comment	<p>The 'type' of a resource can be many different things. In PRISM descriptions, the dc:type element takes values that indicate the style of presentation of the content, such as "Map", "Table", or "Chart". This is in contrast to prism:category, which represents the genre, or stereotypical intellectual content type, of the resource. For example, the genre 'electionResults' can be presented in a map, a table, or a chart.</p> <p>Recommended practice for PRISM implementations is to use a value from Table 6.0 [PRISMCOMP] Controlled Vocabulary of Presentation Styles, expressed as a URI reference. Implementations MUST also be able to handle text values, but interoperation with text values cannot be guaranteed.</p> <p>To describe the physical size or digital file format of the resource, use the dc:format element.</p>
Model #1	
Element Content	URI Reference (empty element)
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
Example	<p>The two examples below show how prism:type, prism:category, and dc:format all describe different aspects of a resource. For brevity, the examples below use relative URI references. Assume that they are within the scope of a base URI declaration:</p> <p>Model #1</p> <pre><dc:type rdf:resource="resourcetype.xml#article"/> <prism:category rdf:resource="category.xml#column"/> <dc:format>text/html</dc:format></pre> <p>Model #2</p> <pre><dc:type>http://idealliance.resourcetype.xml#birdsEye</dc:type> <prism:category rdf:resource="category.xml#photo"/> <dc:format>image/jpeg</dc:format></pre> <p>Model #3</p> <pre><dc:type rdf:parseType="Literal">resourcetype&eql;wormsEye</dc:type> <prism:category rdf:resource="category.xml#photo"/> <dc:format>image/jpeg</dc:format></pre>

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 1.3, 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, 1988.
<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](http://www.w3.org/2001/sw/RDFCore/mediatype-registration) 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](http://www.w3.org/TR/rdf-primer/) 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-nt-20040210/> . [Latest version](#) available at <http://www.w3.org/TR/rdf-nt/> .

[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/>.