



PRISM:
Publishing Requirements for Industry Standard Metadata

PRISM Specification: Modular: Version 2.0

PRISM Compliance

FINAL

2008 02 19

Copyright and Legal Notices

Copyright (c) International Digital Enterprise Alliance, Inc. [IDEAlliance] (2001 – 2008).

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	New in this Version.....	5
4	PRISM Compliance	7
4.1	Constraints on Systems Producing PRISM Compliant Content	7
4.1.1	Producing Profile One Compliant PRISM Content	7
4.1.2	Producing Profile Two Compliant PRISM Content	7
4.1.3	Producing Profile Three Compliant PRISM Content.....	7
4.2	Constraints on Systems Consuming PRISM Compliant Content	7
4.2.1	Consuming Profile Three Compliant PRISM Content.....	8
4.2.2	Consuming Profile Two Compliant PRISM Content	8
4.2.3	Consuming Profile One Compliant PRISM Content	8
4.3	Constraints on PRISM-compliant Content Models.....	8
4.4	Identifying PRISM Content.....	8
4.5	Namespace and Vocabulary Identifiers	8
4.5.1	Date-time.....	9
4.5.2	Locations.....	9
4.5.3	Industrial Sector	10
4.6	PRISM Controlled Vocabularies.....	10
4.7	Identifiers.....	10
4.8	Cardinality and Optionality	10
4.9	Automatic Creation of Inverse Relations.....	11
5	PRISM Profile of the Resource Description Framework	13

5.1	Constraint 1: Top-level Structure of Descriptions	13
5.2	Constraint 2: rdf:aboutEachPrefix Disallowed.....	13
5.3	Further Qualifications	13
5.4	Conventions for Property Values	14
5.5	Convention for In-line Controlled Vocabulary Term Definitions Preferred	14

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/2.0/PRISM_compliance_2.0.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	6/28/05	Kennedy	Enhance element descriptions and examples. Include RDF discussion as per edits to [PRISMPRISMNS]
1.3B	7/13/05	Kennedy	Resolve group comments
1.3 Final	10/01/05	Kennedy	Resolve open industry comments
2.0 Draft A	07/01/07	Kennedy	Prepare document 2.0 for WG review
2.0 Final Draft	07/12/07	Kennedy	Prepare for public comment
2.0 FD w Edits	09/14/07	Kennedy	Prepare for comment resolution
2.0 Final	10/01/07	Kennedy	Comments resolved
2.0 Release	2/19/08	Kennedy	Final for Release

2 PRISM Documentation Structure

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.

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] http://www.prismstandard.org/specifications/2.0/PRISM_introduction_2.0.pdf	Overview, background, purpose and scope of PRISM; examples; contains no normative material.
PRISM Compliance [PRISMCOMP] http://www.prismstandard.org/specifications/2.0/PRISM_compliance_2.0.pdf	Describes two profiles of PRISM compliance for content and systems; includes normative material.
The PRISM Namespace [PRISMPRISMNS] http://www.prismstandard.org/specifications/2.0/PRISM_prism_namespace_2.0.pdf	Describes the elements contained in the PRISM namespace; includes normative material.
The PRISM Subset of the Dublin Core Namespace [PRISMDCNS] http://www.prismstandard.org/specifications/2.0/PRISM_dublin_core_namespace_2.0.pdf	Describes the elements from the Dublin Core namespace that are included in PRISM; includes normative material.
The PRISM Inline Markup Namespace [PRISMIMNS] http://www.prismstandard.org/specifications/2.0/PRISM_inline_markup_namespace_2.0.pdf	Describes the elements contained in the PRISM Inline Markup Namespace; includes normative material.
The PRISM Rights Language Namespace [PRISMRLNS] http://www.prismstandard.org/specifications/2.0/PRISM_rights_namespace_2.0.pdf	Describes the elements contained in the PRISM Rights Language Namespace; includes normative material.
The PRISM Controlled Vocabulary Namespace [PRISMCVNS] http://www.prismstandard.org/specifications/2.0/PRISM_controlled_vocabulary_namespace_2.0.pdf	Describes the elements contained in the PRISM Controlled Vocabulary Namespace; includes normative material. The PRISM Controlled Vocabularies are now documented in this document.
The PRISM Aggregator Message Namespace [PRISMAMNS] http://www.prismstandard.org/specifications/2.0/PRISM_prism_aggregator_message_namespace_2.0.pdf	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), an XML-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 Message V. 2.0 \[PAMGUIDE\]](#).

2.2.2 Access to PRISM Documentation

The PRISM documentation package, the PAM guide (see above), the PAM DTD, the PAM XSD 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 required or normative aspects of the PRISM Specification, with reference to content and systems that wish to assert that they are "PRISM compliant".

Since the PRISM Specification, per se, does not require a specific machine-verifiable format -- there is no PRISM DTD or schema -- it is possible to make use of PRISM elements in a wide range of ways, not all of which will provide the benefit of standardized content that PRISM was designed to realize. Consequently, the PRISM Working Group has identified two general profiles of compliance, described in this document. Adherence to either one will provide a reliable framework for metadata within PRISM's area of applicability.

3.2 New in this Version

See PRISM Introduction 2.0 [PRISMINT] for all changes.

Changes in this document include:

- Added Profile Three for PRISM XMP

4 PRISM Compliance

Since PRISM is designed to specify the form of content maintained in and exchanged between systems, it does not set out to constrain the behavior of systems to any greater extent than necessary. It also recognizes a different set of constraints upon systems when they are producing PRISM-compliant content and when they are consuming it. And, as described below, it provides for two different forms of PRISM compliance in the content itself.

In an effort to provide the maximum utility to those who adopt PRISM, the PRISM Working Group has defined three different forms of PRISM compliance: **profile three**, **profile two** and **profile one**. The intent of these forms is to ensure that a system receiving PRISM compliant content can rely on the meaning of metadata as specified in this document, and, if the content is asserted to be profile two compliant, that it will also be structured as specified here. The PRISM Working Group recommends selecting a compliance profile based on individual business requirements for PRISM.

Finally, a system that claims either profile of PRISM compliance is only constrained in a very minimal way with regard to the specific PRISM namespaces and elements it supports. Consequently, the developers of such systems **MUST** publish and maintain an accurate description of the PRISM namespaces and elements they support, in order to claim either form of "PRISM compliance" under the terms of this specification.

4.1 Constraints on Systems Producing PRISM Compliant Content

Systems **MUST** assert that they are capable of producing profile three, profile two or profile one PRISM compliant content. At this point, the assertion is assumed to be contractual, not machine readable. Content of any supported PRISM elements **MUST** be as described in this specification. Specifically, systems **MUST NOT** add elements or attributes to PRISM namespaces and vocabularies or to the Dublin Core namespace; systems **MUST NOT** define optional elements as mandatory.

4.1.1 Producing Profile One Compliant PRISM Content

A profile one compliant system **MUST** produce content structured in well-formed XML. It **MUST** support the Dublin Core namespace and the dc:identifier element. It **MUST** support one or more PRISM namespaces and one or more elements from each supported namespace.

4.1.2 Producing Profile Two Compliant PRISM Content

A profile two compliant system **MUST** produce content structured as specified in [Section 5: PRISM Profile of the Resource Description Framework](#). It **MUST** support the Dublin Core namespace, the RDF namespace, rdf:ID, and the rdf:about element, but their use is not required. It **MUST** support one or more PRISM namespaces and one or more elements from each supported namespace.

4.1.3 Producing Profile Three Compliant PRISM Content

A profile three compliant system **MUST** produce XMP metadata according to the "[XMP Storage Model](#)" as specified in the [Adobe Extensible Metadata Platform \(XMP\) Specification \[XMP\]](#). This includes the serialization of the metadata as a stream of XML and XMP Packets, a means of packaging the data in files. It **MUST** support the Dublin Core namespace and the dc:identifier element. It **MUST** support one or more PRISM namespaces.

4.2 Constraints on Systems Consuming PRISM Compliant Content

Systems **MUST** assert that they are capable of consuming profile three, profile two or profile one PRISM-compliant content. At this point, the assertion is assumed to be contractual, not machine readable. Systems **MUST** treat content of any supported PRISM elements as described in this specification. Specifically, systems **MUST NOT** add elements or attributes to PRISM namespaces and vocabularies or to the Dublin Core namespace; systems **MUST NOT** define optional elements as mandatory.

Systems are not required to discard well-formed metadata that is unknown or not interpretable within their scope. Systems **SHOULD** retain and retransmit any information that is not malformed or otherwise non-compliant, regardless of its utility or value within their scope.

Systems MUST be capable of handling elements and attributes that are not part of the PRISM Specification without generating an error. A well-formed element or attribute, otherwise unknown, MUST NOT be considered an error in PRISM-compliant content.

4.2.1 Consuming Profile Three Compliant PRISM Content

A profile three compliant system MUST consume ("read," "accept as input") content structured as XMP. It MUST expect the Dublin Core namespace and one or more PRISM namespaces and one or more XMP metadata fields/elements from each supported namespace.

4.2.2 Consuming Profile Two Compliant PRISM Content

A profile two compliant system MUST consume ("read," "accept as input") content structured as specified in [Section 5: PRISM Profile of the Resource Description Framework](#). It MUST expect the Dublin Core namespace, rdf:ID, and the rdf:about element; It MUST support one or more PRISM namespaces and one or more elements from each supported namespace.

4.2.3 Consuming Profile One Compliant PRISM Content

A profile one compliant system MUST consume ("read," "accept as input") content structured in well-formed XML. It MUST expect the Dublin Core namespace and the dc:identifier element. It MUST support one or more PRISM namespaces and one or more elements from each supported namespace.

4.3 Constraints on PRISM-compliant Content Models

Where an organization is developing a content model using PRISM namespaces and elements, whether or not the resulting abstraction of content (a set of database tables, a specification, a DTD, a schema, etc.) is realized as part of a software system, if PRISM compliance is to be claimed, then the same constraints apply to the content models as apply to PRISM content producers.

4.4 Identifying PRISM Content

The Internet Media Type (aka MIME type)[IETF-MIMETYPES] for profile two compliant PRISM descriptions is "text/prism+rdf+xml". The Internet Media Type for profile one compliant PRISM descriptions is "text/prism+xml". Because PRISM/XMP may be embedded in a wide variety of media types, this profile will assume the media type of the resource. PRISM Best Practice is to specify an added "prism+xml". So, for example, if the resource is a PDF, the mime type would be "application/pdf+prism+xml".

When PRISM descriptions are stored as profile two or profile one XML files, the preferred filename extension is ".prism". When neither of those two identification methods are appropriate, the content can be scanned for occurrences of the URI "http://www.prismstandard.org/namespaces/2.0/basic" used as a namespace URI in an XML documents. Such documents are considered to be PRISM content.

When PRISM descriptions are stored as profile three, the metadata will be stored inside each media object within the XMP Packet. If exported, the PRISM XMP fields are stored within the exported *.xmp files. PRISM metadata within an .xmp file can be identified by the dc: or prism: namespaces attached to the field.

4.5 Namespace and Vocabulary Identifiers

Systems that claim PRISM profile two or one compliance MUST recognize and support namespaces as defined in [sections 4.1](#) and [4.2](#). Systems MAY use the namespace declarations below in order to use familiar prefixes.

Namespace	Recommended Namespace Declaration
Resource Description Framework	xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
Dublin Core	xmlns:dc="http://purl.org/dc/elements/1.1"
PRISM	xmlns:prism="http://prismstandard.org/namespaces/2.0/basic/

PRISM Controlled Vocabulary	xmlns:pcv="http://prismstandard.org/namespaces/2.0/pcv/"
PRISM Inline Markup	xmlns:pim="http://prismstandard.org/namespaces/2.0/pim/"
PRISM Aggregator Message	xmlns:prl="http://prismstandard.org/namespaces/2.0/pam/"

Table 2: Namespaces Used In PRISM Descriptions

The PRISM Specification also defines a number of controlled vocabularies. The base URIs for those vocabularies are:

Vocabulary Name	Base URI
PRISM Aggregation Type	http://prismstandard.org/vocabularies/2.0/aggregationtype.xml
PRISM Compliance Profile	http://prismstandard.org/vocabularies/2.0/complianceprofile.xml
PRISM Genre	http://prismstandard.org/vocabularies/2.0/genre.xml
PRISM Platform	http://prismstandard.org/vocabularies/2.0/platform.xml
PRISM Resource Type	http://prismstandard.org/vocabularies/2.0/resourcetype.xml
PRISM Rights	http://prismstandard.org/vocabularies/2.0/rights.xml
PRISM Role	http://prismstandard.org/vocabularies/2.0/role.xml
PAM Class	http://prismstandard.org/vocabularies/2.0/pam.xml

Table 3.0 Base URIs for PRISM Controlled Vocabularies

In addition to the PRISM-defined vocabularies, a number of other vocabularies and data formats are recommended by PRISM as current best practice. Those are:

4.5.1 Date-time

PRISM-compliant applications sending metadata to other systems are STRONGLY ENCOURAGED to use the W3C profile of ISO 8601 [W3C-DateTime] as the format of their date and time values, including time zone data (tz). Implementers are advised, however, that this specification may be supplanted in the future by one which allows features such as ranges of times, or the use of the tz library's method of specifying time zone offsets as strings composed of Continent/City. So, implementations SHOULD be able to deal with other forms.

4.5.2 Locations

PRISM-compliant applications sending metadata to other systems are STRONGLY ENCOURAGED to use the codes from [ISO-3166] as the values for the <prism:location> element.

ISO has not yet defined a standard URI convention for those codes. In order to maximize interoperability, implementations MAY wish to use the following non-resolvable URLs.

<http://prismstandard.org/vocabs/ISO-3166/XX>

where XX is a 2-letter uppercase country code, and

<http://prismstandard.org/vocabs/ISO-3166-2/XX-YYY>

where XX is as above and YYY is a one to three-character alphanumeric subregion code.

The ISO 3166 codes do not cover cities, counties, or historical locations. In situations where finer coverage is needed, implementers MAY wish to use codes from the Thesaurus of Geographic Names [TGN].

4.5.3 Industrial Sector

PRISM-compliant applications sending metadata to other systems MAY wish to use the industry sector codes from [NAICS] as the values for the <prism:industry> element and <pim:industry>'s href attribute.

4.6 PRISM Controlled Vocabularies

The PRISM Specification generally focuses on the elements and attributes that may be used in a PRISM metadata document. Elements, in effect, define the syntax of the document. To convey the meaning of a document, the values that a given element may take must also be defined. Other elements use controlled vocabularies created and maintained by third parties (such as the ISO 3166 codes for country names). Still other elements will require some domain-specific controlled vocabulary (e.g., the North American Industrial Classification System).

PRISM Controlled Vocabularies are now documented in The PRISM Controlled Vocabulary Namespace [PRISMCVNS]. These have been moved from this document to the PRISM Controlled Vocabulary Namespace to assist end users in locating the controlled vocabularies easily.

4.7 Identifiers

PRISM profile two compliant files MUST use the rdf:about attribute on rdf:Description elements to specify the resource being described. The value of the rdf:about attribute is STRONGLY ENCOURAGED to be a URI reference [RFC-2396]. The dc:identifier element MUST be used to contain any additional identifiers to be sent, or any identifiers that cannot be represented as a URI reference. For example, a resource can be identified by a URI and by an internal asset ID that an organization would use to access it in their database. PRISM-compliant applications are STRONGLY ENCOURAGED to maintain the unique identifier(s) provided for a resource.

PRISM profile one compliant files MUST use the dc:identifier element to specify the resource being described. This value is STRONGLY ENCOURAGED to be a unique identifier.

PRISM's only policy on the assignment of identifiers is that the party assigning an identifier MUST NOT assign the same identifier to a different resource, using whatever definition of 'different' the assigning party deems appropriate.

PRISM compliant systems MUST regard two resources as being 'the same' if they have the same unique identifier. The party assigning the identifier is the sole arbiter of what they mean by 'the same'. Note that this definition does not imply that two resources are different if their identifiers are different. Different identifiers MAY (and frequently will) be assigned to the same resource.

PRISM does not require that all resources carry the same identifier through their entire lifecycle. However, if the publisher assigns a new identifier to non-reusable content obtained from an external party, the publisher SHOULD retain information on the origin and licensing of the resource so that someone later in its lifecycle can determine how to obtain the rights to reuse it.

4.8 Cardinality and Optionality

All PRISM descriptions MUST contain at least one identifier for the resource being described, expressed in the rdf:about attribute (profile two) or the dc:identifier element (profile one). Any number of additional identifiers MAY be expressed in dc:identifier elements. However, at least one other field MUST be specified in a description in order to have a meaningful model.

All other Dublin Core elements are optional, and any of them MAY be repeated any number of times. Unless specifically noted otherwise, PRISM elements are also optional and MAY occur any number of times in a description.

4.9 Automatic Creation of Inverse Relations

PRISM includes elements for specifying relations between resources (e.g. Resource1 isVersionOf Resource2). Those relations have inverse relations that are also in the PRISM Specification (e.g., Resource2 hasVersion Resource1).

PRISM-compliant systems which receive one side of such a relation MAY infer the presence of the additional inverse relation. To be more specific, if the implementation tracks the origin of individual RDF statements and can segregate its database in order to undo the addition of such inferred inverses, it SHOULD infer the inverse and keep it segregated from the original input. If an implementation does not track individual statements and sources, it MAY infer the inverse relations but is cautioned about the possibility of data corruption.

5 PRISM Profile of the Resource Description Framework

The Resource Description Framework (RDF) has been standardized by the W3C to provide a general framework for metadata. As such, its capabilities exceed those required by PRISM. Therefore, this document specifies a 'profile' – a restricted subset – of RDF that all PRISM profile two-compliant software MUST support. This profile excludes certain capabilities of RDF that are not needed in PRISM applications, thus simplifying the development of PRISM applications.

Applications conforming to the PRISM Specification and claiming profile two compliance MUST produce correct RDF documents that can be read by any RDF-compliant software. They MUST also produce documents that conform to the PRISM profile of RDF. PRISM-compliant software does not have to be capable of processing arbitrary RDF documents.

5.1 Constraint 1: Top-level Structure of Descriptions

The formal grammar for RDF [W3C-RDF] specifies:

[6.1] RDF ::= ['<rdf:RDF>'] obj* ['</rdf:RDF>']

[6.2] obj ::= description | container

For PRISM descriptions, the rdf:RDF wrapper element is required, and its child elements are restricted to being rdf:Description elements. The production that replaces productions 6.1 and 6.2 for PRISM systems is:

RDF ::= '<rdf:RDF' namespace_decls '>' description+ '</rdf:RDF>'

5.2 Constraint 2: rdf:aboutEachPrefix Disallowed

PRISM descriptions MUST NOT use the rdf:aboutEachPrefix attribute. Production [6.8] of the RDF M&S specification thus becomes:

AboutEachAttr ::= ' aboutEach="" URI-reference ""'

5.3 Further Qualifications

No other overall restrictions in the allowed RDF syntax are specified in this section. However, implementers are advised to pay particular attention to the following points:

Many elements, such as dc:subject, may take a string as a value, or may use a URI for identifying an element in a controlled vocabulary of subject description codes. The URI may be a simple reference, or may provide an inline description of the controlled vocabulary term. Implementations MUST be capable of handling all three of those cases reliably.

Implementers must decide how their system will deal with unsupported descriptive elements. The PRISM Specification does not preclude other descriptive elements, although their interoperability cannot be guaranteed. PRISM implementations MAY retain unknown descriptive elements and retransmit them.

To aid automated processing of PRISM metadata, this specification defines a separate namespace for PRISM elements suitable for in-line markup. Thus, prism:organization is an RDF statement and pim:organization is used as in-line markup.

The PRISM Working Group encourages implementers to keep the generated markup as simple as possible. As an example, if a work has multiple authors, RDF allows that situation to be encoded in two ways, which have slightly different meanings. The first way uses multiple dc:creator elements, each listing a separate author. The second way is to have a single dc:creator element, which then contains one of RDF's collection constructs, such as rdf:Bag. That, in turn, would list the different authors. According to the RDF specification, the first is to be used when the authors acted as a collection of individuals in the creation of a work. The second is to be used when the authors acted as a committee. Experience has shown, however, that this distinction is too subtle for human catalogers to make reliably. The PRISM Working Group recommends using the first approach in most cases.

Note that although a sequence of dc:creator elements in an RDF/XML file implicitly defines a sequence (in the XML world), RDF parsers have no obligation to preserve that ordering, unlike if an explicit rdf:Seq were given. PRISM implementers are advised that there are quality of implementation issues between different RDF processors. In general, implementers MAY prefer to build on top of an RDF parser that allows the original order of the statements to be reconstructed. That would allow the original order of the authors on a piece to be reconstructed, which might or might not convey additional meaning to the viewer of a styled version of the record. Similarly, XML software that can handle the recently-standardized xml:base attribute MAY be preferred.

5.4 Conventions for Property Values

To aid in the automatic processing of PRISM documents, PRISM utilizes some conventions in expressing values of RDF properties. The values are expressed in three ways. First, a resource or an entry in a controlled vocabulary MAY be referenced with the rdf:resource attribute. For example, a book can be identified by its ISBN number as follows:

```
<dc:identifier rdf:resource="urn:isbn:0-932592-00-7"/>
```

Second, human readable text MUST be represented as element content:

```
<dc:title>Juggling for the Complete Klutz</dc:title>
```

barring any circumstances where representing the text in element content would change the RDF as compared to representing it as an attribute value. That element content may contain XML markup, in which case the rdf:parseType attribute MUST be given and MUST have a value of 'Literal'.

Third, controlled vocabulary entries may be specified in-line. See Example 1:

```
<dc:subject>
  <pcv:Descriptor rdf:about="http://loc.gov/LC/QA-76">
    <pcv:vocabulary>Library of Congress Classification</pcv:vocabulary>
    <pcv:code>QA-76</pcv:code>
    <pcv:label>Mathematical software</pcv:label>
  </pcv:Descriptor>
</dc:subject>
```

Example 1: In-Line Controlled Vocabulary Entries

XML DTDs cannot describe such a flexible content model, but more recent schema languages such as XML Schema and RELAX can, with varying degrees of difficulty.

5.5 Convention for In-line Controlled Vocabulary Term Definitions Preferred

PRISM descriptions make extensive use of values selected from controlled vocabularies. Conceptually, all that is needed is a reference to the vocabulary entry. But for practical considerations such as human readability, ease of use of full-text search tools, and performance, it is useful to be able to provide information about the controlled vocabulary entry, such as its human-readable label, directly in the description.

The PRISM Specification recommends that when this additional information is provided, that it be provided in-line, instead of as an additional rdf:Description element. In Example 2, a story whose subject is "Mining" as defined in the North American Industrial Classification System (NAICS), would have the following description:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:prism="http://prismstandard.org/namespaces/2.0/basic/"
  xmlns:pcv="http://prismstandard.org/namespaces/pcv/2.0/"
  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="story.xml">
    <dc:subject>
      <pcv:Descriptor rdf:about="http://prismstandard.org/vocabs/NAICS/21">
        <pcv:vocab>North American Industrial Classification System</pcv:vocab>
        <pcv:code>21</pcv:code>
        <pcv:label>Mining</pcv:label>
      </pcv:Descriptor>
    </dc:subject>
  </rdf:Description>
</rdf:RDF>
```

Example 2: In-Line Description

as opposed to the form of the description in Example 3, where the controlled vocabulary term is described out-of-line instead of in-line.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:prism="http://prismstandard.org/namespaces/2.0/basic/"
  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="story.xml">
    <dc:subject rdf:resource="http://prismstandard.org/vocabs/NAICS/21"/>
  </rdf:Description>

  <pcv:Descriptor rdf:about="http://prismstandard.org/vocabs/NAICS/21">
    <pcv:vocab>North American Industrial Classification System</pcv:vocab>
    <pcv:code>21</pcv:code>
    <pcv:label>Mining</pcv:label>
  </pcv:Descriptor>
</rdf:RDF>
```

Example 3: Out-of-line Description

The two approaches are identical in terms of the RDF graph that is generated, but the former is believed easier to deal with using standard tools such as full-text indexing software or simple editing scripts.

Note that we use the `rdf:about` attribute when providing the information on the controlled vocabulary term. This indicates that the real definition of the term is elsewhere, and we are merely providing some local descriptions of that term