



Digital Music, Photo, and Video Collections



Dublin Core-Normalized Metadata Format Profile Specification

Revision 1.01

11 March 2003

© 2002-2003 Optical Storage Technology Association

This document is available at <http://www.osta.org/mpv/public/specs/DC-NMF-Profile-Spec-1.01.PDF>

POINTS OF CONTACT

<p><u>OSTA</u> David Bunzel OSTA President</p> <p>Tel: +1 (408) 253-3695 Email: dbunzel@osta.org</p> <p>http://www.osta.org</p> <p><u>MPV Website</u></p> <p>http://www.osta.org/mpv/</p>	<p><u>Technical Content</u></p> <p>Gabe Beged-Dov Editor, Normalized Metadata Format Specification</p> <p>Tel: +1 541-715-7347 Email: Gabe_Beged-Dov@hp.com</p> <p>Pieter van Zee Editor, MPV Specification MPV Initiative Lead</p> <p>Tel: +1 541-715-8658 Email: Pieter_van_Zee@hp.com</p> <p>Felix Nemirovsky Chairman, MultiRead Subcommittee</p> <p>Tel: +1 415 643 0944 Email: felixn@pacbell.net</p>
--	---

ABSTRACT

The Dublin Core Normalized Metadata Format Profile (DC-NMF) specifies the NMF representation of Dublin Core Metadata Initiative compliant metadata.

COPYRIGHT NOTICE

Copyright 2002-2003 Optical Storage Technology Association, Inc.

RELEASE HISTORY

<i>Version</i>	<i>Date</i>	<i>Comments</i>
1.00	23 October 2002	First public release.
1.01	11 March 2003	Schema and document format is unchanged. Changed name from MultiPhoto/Video to MPV and MusicPhotoVideo; updated logo, contact info, copyright. Added Music Profile mention. Updated graphics to use MPV and Music Profile. No other changes.

LICENSING IMPORTANT NOTICES

- (a) THIS DOCUMENT IS AN AUTHORIZED AND APPROVED PUBLICATION OF THE OPTICAL STORAGE TECHNOLOGY ASSOCIATION (OSTA). THE SPECIFICATIONS CONTAINED HEREIN ARE THE EXCLUSIVE PROPERTY OF OSTA BUT MAY BE REFERRED TO AND UTILIZED BY THE GENERAL PUBLIC FOR ANY LEGITIMATE PURPOSE, PARTICULARLY IN THE DESIGN AND DEVELOPMENT OF WRITABLE OPTICAL SYSTEMS AND SUBSYSTEMS. THIS DOCUMENT MAY BE COPIED IN WHOLE OR IN PART PROVIDED THAT NO REVISIONS, ALTERATIONS, OR CHANGES OF ANY KIND ARE MADE TO THE MATERIALS CONTAINED HEREIN.
- (b) COMPLIANCE WITH THIS DOCUMENT MAY REQUIRE USE OF ONE OR MORE FEATURES COVERED BY THE PATENT RIGHTS OF AN OSTA MEMBER, ASSOCIATE OR THIRD PARTY. NO POSITION IS TAKEN BY OSTA WITH RESPECT TO THE VALIDITY OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT, WHETHER OWNED BY A MEMBER OR ASSOCIATE OF OSTA OR OTHERWISE. OSTA HEREBY EXPRESSLY DISCLAIMS ANY LIABILITY FOR INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS OF OTHERS BY VIRTUE OF THIS OSTA DOCUMENT, NOR DOES OSTA UNDERTAKE A DUTY TO ADVISE USERS OR POTENTIAL USERS OF OSTA DOCUMENTS OF SUCH NOTICES OR ALLEGATIONS. OSTA HEREBY EXPRESSLY ADVISES ALL USERS OR POTENTIAL USERS OF THIS DOCUMENT TO INVESTIGATE AND ANALYZE ANY POTENTIAL INFRINGEMENT SITUATION, SEEK THE ADVICE OF INTELLECTUAL PROPERTY COUNSEL AND, IF INDICATED, OBTAIN A LICENSE UNDER ANY APPLICABLE INTELLECTUAL PROPERTY RIGHT OR TAKE THE NECESSARY STEPS TO AVOID INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT. OSTA EXPRESSLY DISCLAIMS ANY INTENT TO PROMOTE INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT BY VIRTUE OF THE EVOLUTION, ADOPTION, OR PUBLICATION OF THIS OSTA DOCUMENT.
- (c) ONE OR MORE PATENT HOLDERS HAVE FILED STATEMENTS OF WILLINGNESS TO GRANT A LICENSE, ON REASONABLE AND NONDISCRIMINATORY TERMS, ON A RECIPROCAL BASIS, UNDER PATENT CLAIMS ESSENTIAL TO IMPLEMENT THIS SPECIFICATION. FURTHER INFORMATION MAY BE OBTAINED FROM OSTA.
- (d) OSTA MAKES NO REPRESENTATION OR WARRANTY REGARDING ANY SPECIFICATION, AND ANY COMPANY USING A SPECIFICATION SHALL DO SO AT ITS SOLE RISK, INCLUDING SPECIFICALLY THE RISKS THAT A PRODUCT DEVELOPED WILL NOT BE COMPATIBLE WITH ANY OTHER PRODUCT OR THAT ANY PARTICULAR PERFORMANCE WILL NOT BE ACHIEVED. OSTA SHALL NOT BE LIABLE FOR ANY EXEMPLARY, INCIDENTAL, PROXIMATE OR CONSEQUENTIAL DAMAGES OR EXPENSES ARISING FROM THE USE OR IMPLEMENTATION OF THIS DOCUMENT. THIS DOCUMENT DEFINES ONLY ONE APPROACH TO COMPATIBILITY, AND OTHER APPROACHES MAY BE AVAILABLE IN THE INDUSTRY.
- (e) THIS DOCUMENT IS A SPECIFICATION ADOPTED BY OSTA. THIS DOCUMENT MAY BE REVISED BY OSTA AT ANY TIME AND WITHOUT NOTICE AND USERS ARE ADVISED TO OBTAIN THE LATEST VERSION. IT IS INTENDED SOLELY AS A GUIDE FOR ORGANIZATIONS INTERESTED IN DEVELOPING PRODUCTS WHICH CAN BE COMPATIBLE WITH OTHER PRODUCTS DEVELOPED USING THIS DOCUMENT. THIS DOCUMENT IS PROVIDED "AS IS".
- (f) THE MPV NAME, THE MPV LOGO, AND THE MusicPhotoVideo NAME ARE TRADEMARKS OF OPTICAL STORAGE TECHNOLOGY ASSOCIATION, INC. ALL OTHER TRADEMARKS ARE THE PROPERTY OF THEIR RESPECTIVE OWNERS. THESE OSTA TRADEMARKS MAY NOT BE USED EXCEPT FOR JOURNALISTIC PURPOSES WITHOUT AN EXPLICIT LICENSE FROM OSTA.

Contents

Contents.....	4
Chapter 1: Introduction.....	9
1.1 Terminology.....	9
1.1.1 Requirements.....	10
1.2 Dublin Core Background Information.....	10
1.3 Schema Information.....	10
1.4 DC-NMF Best Practices and Usage Guidelines.....	10
1.4.1 Capitalization	11
1.4.2 Property Usage.....	11
1.4.3 Property Types.....	11
1.4.4 Plain-text vs. XML markup	12
1.4.5 Qualified Properties and Dumb-Down.....	12
1.4.6 Array Properties.....	13
Chapter 2: Dublin Core Elements Schema.....	14
2.1 Best Practices and Usage.....	14
2.2 Properties container.....	14
2.3 Property: contributor	16
2.3.1 group contributorChoiceGroup.....	16
2.3.2 element contributor.....	17
2.3.3 element contributorBag	17
2.3.4 element contributorQVal.....	17
2.3.5 element contributorRef	18
2.4 Property: coverage.....	18
2.4.1 group coverageChoiceGroup.....	19
2.4.2 element coverage	19
2.4.3 element coverageQVal.....	19
2.4.4 element coverageRef	20
2.5 Property: creator	20
2.5.1 group creatorChoiceGroup.....	20
2.5.2 element creator.....	21
2.5.3 element creatorRef	21
2.5.4 element creatorSeq	21
2.6 Property: date.....	22
2.6.1 group dateChoiceGroup.....	22
2.6.2 element date	22
2.6.3 element dateRef	23
2.7 Property: description.....	23
2.7.1 group descriptionChoiceGroup	23
2.7.2 element description	24
2.7.3 element descriptionAlt.....	24
2.7.4 element descriptionAnyXML	24
2.7.5 element descriptionRef.....	25
2.8 Property: format.....	25
2.8.1 group formatChoiceGroup.....	25

2.8.2	element format.....	26
2.8.3	element formatRef	26
2.9	Property: identifier.....	26
2.9.1	group identifierChoiceGroup	27
2.9.2	element identifier.....	27
2.9.3	element identifierRef.....	27
2.10	Property: language.....	27
2.10.1	group languageChoiceGroup.....	28
2.10.2	element language.....	28
2.10.3	element languageBag	29
2.10.4	element languageQVal.....	29
2.10.5	element languageRef.....	30
2.11	Property: publisher.....	30
2.11.1	group publisherChoiceGroup.....	30
2.11.2	element publisher.....	31
2.11.3	element publisherBag.....	31
2.11.4	element publisherQVal.....	31
2.11.5	element publisherRef.....	32
2.12	Property: relation.....	32
2.12.1	group relationChoiceGroup.....	32
2.12.2	element relation	33
2.12.3	element relationBag.....	33
2.12.4	element relationQVal.....	34
2.12.5	element relationRef	34
2.13	Property: rights.....	34
2.13.1	group rightsChoiceGroup.....	35
2.13.2	element rights	35
2.13.3	element rightsAlt.....	35
2.13.4	element rightsQVal.....	36
2.13.5	element rightsRef.....	36
2.14	Property: source.....	36
2.14.1	group sourceChoiceGroup.....	37
2.14.2	element source	37
2.14.3	element sourceQVal.....	38
2.14.4	element sourceRef	38
2.15	Property: subject.....	38
2.15.1	group subjectChoiceGroup.....	39
2.15.2	element subject.....	39
2.15.3	element subjectBag.....	39
2.15.4	element subjectQVal.....	40
2.15.5	element subjectRef.....	40
2.16	Property: title.....	41
2.16.1	group titleChoiceGroup.....	41
2.16.2	element title.....	41
2.16.3	element titleAlt	42
2.16.4	element titleAnyXML.....	42
2.16.5	element titleQVal	43
2.16.6	element titleRef.....	43
2.17	Property: type.....	43
2.17.1	group typeChoiceGroup.....	44
2.17.2	element type	44
2.17.3	element typeBag.....	44
2.17.4	element typeQVal.....	45
2.17.5	element typeRef.....	45
Chapter 3:	Dublin Core Terms Schema.....	46
3.1	Best Practices and Usage.....	46
3.2	Properties Container.....	46
3.3	Property: abstract.....	48
3.3.1	group abstractChoiceGroup.....	48
3.3.2	element abstract.....	49
3.3.3	element abstractAlt.....	49
3.3.4	element abstractAnyXML.....	50

3.3.5	element abstractQVal.....	50
3.3.6	element abstractRef	50
3.4	Property: alternative	51
3.4.1	group alternativeChoiceGroup	51
3.4.2	element alternative	51
3.4.3	element alternativeAlt.....	52
3.4.4	element alternativeAnyXML	52
3.4.5	element alternativeQVal.....	52
3.4.6	element alternativeRef	53
3.5	Property: available	53
3.5.1	group availableChoiceGroup	53
3.5.2	element available.....	54
3.5.3	element availableRef.....	54
3.6	Property: conformsTo	54
3.6.1	group conformsToChoiceGroup	55
3.6.2	element conformsTo.....	55
3.6.3	element conformsToBag.....	55
3.6.4	element conformsToQVal	56
3.6.5	element conformsToRef.....	56
3.7	Property: created.....	56
3.7.1	group createdChoiceGroup.....	57
3.7.2	element created	57
3.7.3	element createdRef	57
3.8	Property: extent.....	57
3.8.1	group extentChoiceGroup	58
3.8.2	element extent	58
3.8.3	element extentRef.....	58
3.9	Property: hasFormat.....	59
3.9.1	group hasFormatChoiceGroup	59
3.9.2	element hasFormat.....	60
3.9.3	element hasFormatBag.....	60
3.9.4	element hasFormatQVal.....	60
3.9.5	element hasFormatRef.....	61
3.10	Property: hasPart	61
3.10.1	group hasPartChoiceGroup	61
3.10.2	element hasPart.....	62
3.10.3	element hasPartBag	62
3.10.4	element hasPartQVal	63
3.10.5	element hasPartRef.....	63
3.11	Property: hasVersion	63
3.11.1	group hasVersionChoiceGroup	64
3.11.2	element hasVersion.....	64
3.11.3	element hasVersionBag	64
3.11.4	element hasVersionQVal.....	65
3.11.5	element hasVersionRef.....	65
3.12	Property: isformatOf.....	65
3.12.1	group isFormatOfChoiceGroup.....	66
3.12.2	element isFormatOf.....	66
3.12.3	element isFormatOfBag	66
3.12.4	element isFormatOfQVal.....	67
3.12.5	element isFormatOfRef	67
3.13	Property: isPartOf.....	67
3.13.1	group isPartOfChoiceGroup.....	68
3.13.2	element isPartOf.....	68
3.13.3	element isPartOfBag	68
3.13.4	element isPartOfQVal.....	69
3.13.5	element isPartOfRef	69
3.14	Property: isReferencedBy.....	69
3.14.1	group isReferencedByChoiceGroup.....	70
3.14.2	element isReferencedBy.....	70
3.14.3	element isReferencedByBag	70
3.14.4	element isReferencedByQVal.....	71

3.14.5	element isReferencedByRef	71
3.15	isReplacedBy	71
3.15.1	group isReplacedByChoiceGroup	72
3.15.2	element isReplacedBy	72
3.15.3	element isReplacedByBag	73
3.15.4	element isReplacedByQVal	73
3.15.5	element isReplacedByRef	73
3.16	Property: isRequiredBy	74
3.16.1	group isRequiredByChoiceGroup	74
3.16.2	element isRequiredBy	74
3.16.3	element isRequiredByBag	75
3.16.4	element isRequiredByQVal	75
3.16.5	element isRequiredByRef	75
3.17	Property: issued	76
3.17.1	group issuedChoiceGroup	76
3.17.2	element issued	76
3.17.3	element issuedRef	77
3.18	Property: isVersionOf	77
3.18.1	group isVersionOfChoiceGroup	77
3.18.2	element isVersionOf	78
3.18.3	element isVersionOfBag	78
3.18.4	element isVersionOfQVal	78
3.18.5	element isVersionOfRef	79
3.19	Property: medium	79
3.19.1	group mediumChoiceGroup	79
3.19.2	element medium	80
3.19.3	element mediumRef	80
3.20	Property: modified	80
3.20.1	group modifiedChoiceGroup	80
3.20.2	element modified	81
3.20.3	element modifiedRef	81
3.21	Property: references	81
3.21.1	group referencesChoiceGroup	82
3.21.2	element references	82
3.21.3	element referencesBag	82
3.21.4	element referencesQVal	83
3.21.5	element referencesRef	83
3.22	Property: replaces	83
3.22.1	group replacesChoiceGroup	84
3.22.2	element replaces	84
3.22.3	element replacesBag	84
3.22.4	element replacesQVal	85
3.22.5	element replacesRef	85
3.23	Property: requires	85
3.23.1	group requiresChoiceGroup	86
3.23.2	element requires	86
3.23.3	element requiresBag	86
3.23.4	element requiresQVal	87
3.23.5	element requiresRef	87
3.24	Property: spatial	87
3.24.1	group spatialChoiceGroup	88
3.24.2	element spatial	88
3.24.3	element spatialQVal	88
3.24.4	element spatialRef	89
3.25	Property: tableOfContents	89
3.25.1	group tableOfContentsChoiceGroup	89
3.25.2	element tableOfContents	90
3.25.3	element tableOfContentsAlt	90
3.25.4	element tableOfContentsAnyXML	90
3.25.5	element tableOfContentsQVal	91
3.25.6	element tableOfContentsRef	91
3.26	Property: temporal	91

3.26.1	group temporalChoiceGroup.....	92
3.26.2	element temporal.....	92
3.26.3	element temporalQVal.....	92
3.26.4	element temporalRef.....	93
3.27	Property: valid.....	93
3.27.1	group validChoiceGroup.....	93
3.27.2	element valid.....	94
3.27.3	element validRef.....	94
Appendix 1.	References.....	95

Chapter 1: Introduction

This specification defines a profile based on the Normalized Metadata Format (NMF [NMF]) for metadata that is compatible with the Dublin Core Metadata Initiative [DCMI]. This profile is referred to as the DC-NMF profile when there is a need to distinguish it from other representations of DCMI compatible metadata.

1.1 Terminology

Resource

A Resource is anything that can be identified by a URI

Schema

A Schema is a set of property definitions that is identified by a URI.

Property

A Property is a named entry in a schema. In NMF, this property must have the same namespace as the schema and also have a well-defined value type that is expressible in an XML Schema [XSCHEMA].

Statement

A statement is the binding of a property instance to a particular resource.

Literal Property Value

A literal property value is either textual content or well-formed XML that is not a composite property value.

Composite Property Value

A composite property value is a set of properties from one or more schemas.

Top-level Composite Property Value

A top-level composite property value is a composite property value that is not contained in a property element.

Nested Composite Property Value

A nested composite property value is a composite property value that is contained in a property element.

Array Property Value

An Array property value is sequence of properties that are either ordered, unordered or alternatives.

Qualified Property Value

A qualified property value is one where the base property value (either literal or composite) has zero or more additional properties associated with it. These additional properties are called qualifiers of the base property and provide additional information about how to interpret the base property.

1.1.1 Requirements

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, if and where they appear in this document, are to be interpreted as described in [RFC2119].

1.2 Dublin Core Background Information

The Dublin Core Metadata Initiative is an open forum engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models. DCMI's activities include consensus-driven working groups, global workshops, conferences, standards liaison, and educational efforts to promote widespread acceptance of metadata standards and practices.

The most familiar aspect of the work at the DCMI is the Dublin Core Element Set **[DCES]**. This DCES defines a set of fifteen metadata items that can be used to describe a wide range of resources where a resource is defined very broadly as anything that can be identified by a uniform resource identifier **[URI]**.

There are several alternate ways to interchange Dublin Core metadata that are under development at the DCMI. These include the interchange of DC in HTML **[DC-HTML]**, the interchange of DC using the W3C Resource Description Format **[DC-RDF]**, and the interchange of DC using XML **[DC-XML]**.

This specification specifies how to interchange Dublin Core metadata using the Normalized Metadata Format **[NMF]** which is a representation that can interchange with Resource Description format **[RDF]** metadata and can be managed using mainstream XML tools and technologies.

1.3 Schema Information

The DC-NMF specification uses the following schema:

Schema group	Namespace Identifier	Conventional Namespace Prefix
DC Elements	http://purl.org/dc/elements/1.1/	dc:
DC Terms	http://purl.org/dc/terms/	dcterms:

1.4 DC-NMF Best Practices and Usage Guidelines

The reality of Dublin Core metadata usage is that a very wide variety of content models are employed that are described as complying with Dublin Core in one way or another. The DC-NMF profile balances the need to support a useful subset of these while also providing a more concrete set of interoperability guidelines for users of DC-NMF.

The NMF based representation supports a variety of these variations although there are likely to be some examples of metadata that are specified as being Dublin Core compliant that will not be considered compliant by the NMF representation.

The DC-NMF profile is composed of two schemas that correspond to the primary namespaces that have been defined by the DCMI [DCMI-NS]. These include the DC Elements schema and the DC Terms schema. This section provides general guidelines and best practices that are applicable to the profile as a whole. The chapters on DCES and DC Terms provide additional guidelines that are specific to those schemas.

1.4.1 Capitalization

There are many examples of DC usage that employ capitalization of the element names. This can be seen in the examples in the RDF specification [RDF]. The DC-NMF profile only supports the use of lower case element names as specified in the DC specifications.

- DC-NMF compliant properties MUST use the same namespace and localname as specified in the schemas.

1.4.2 Property Usage

The DC-NMF specification does not tightly define nor constrain the usage and values of DC-NMF properties beyond those articulated by the [DCMI]. This provides for the greatest compatibility with existing [DCMI]-based content. However, the recommended best practice for specifications that use DC-NMF is to more tightly constrain its usage as part of that specification. This will significantly enhance interoperability of DC-NMF-based documents between applications using the given specification. An example of this can be found in the MPV Core Specification [MPV-Core].

1.4.3 Property Types

NMF explicitly models the various content models that are used for describing metadata using different property value types. There are two levels of property types in NMF. The first level is referred to as the base level and the second level is the higher-order level that wraps one or more of the property types from the first level.

The first, or base, level of property types are::

- Simple property type (textual content)
- Composite property type (a set of properties).
- XML Literal property type (well-formed XML that isn't interpreted).
- Reference property type (a reference to another resource via a URI).

NOTE: A property cannot be specified as supporting both a composite property type and a simple property type.

In addition to the base types, NMF supports several higher-order property types that can wrap the base property types. The higher-order property types are:

- Qualified properties
- Array properties

The DC-NMF profile goes out of its way to support most of the property value types on each property. This is done in order to maximize interoperability with metadata that is specified using the RDF interchange representation of DC [DC-RDF].

As mentioned above, NMF does not allow a property to have both a simple and a composite property type. The working assumption for DC-NMF is that most metadata will be based on the use of simple property values. Given this, the DC-NMF profile only supports simple property types for all properties. In other words, DC-NMF property values cannot directly contain subproperties.

Note that this is only relevant to the simple vs. composite choice. A property can still support the other base types such as XML Literal and Reference in addition to the simple property types.

For example, the following cannot be encoded in DC-NMF such that it can be mechanically validated because XML-Schema can only support either textual or element content for the same element name. In DC-NMF, the decision was made to only support textual content.

```
...
<dc:creator>Pieter van Zee</dc:creator>
<dc:creator>
  <rdf:Description>
    <vcard:fname>Gabe</vcard:fname>
    <vcard:lname>Begeed-Dov</vcard:lname>
  </rdf:Description>
```

```
</dc:creator>
...
```

If an application wishes to specify that the value of a DC-NMF property as a composite property, it can do this indirectly on some properties that support the Ref property type. The Ref property type can refer to a top-level composite property which will contain the desired composite property value. Using this approach, the above example can be re-stated as follows.

```
...
<dc:creator>Pieter van Zee</dc:creator>
<dc:creator rdf:resource="#P1"/>

<rdf:Description rdf:about="#P1">
  <vcard:fname>Gabe</vcard:fname>
  <vcard:lname>Begeg-Dov</vcard:lname>
</rdf:Description>
...
```

1.4.4 Plain-text vs. XML markup

There are several DC properties whose contents are intended for presentation to human readers. Examples of these are the title and description properties. These properties are mainly intended to contain plain text but there are many examples of applications placing markup (HTML and/or XML) into these properties. NMF directly supports this type of usage by providing a variant property type that can contain well-formed XML.

- An application SHOULD use the simple property type and plain text for encoding human readable information whenever possible.
- An application SHOULD use the AnyXML variant of a property to convey well-formed markup rather than the base property type.

1.4.5 Qualified Properties and Dumb-Down

Dublin Core Qualifiers [DCQ] supports the use of qualified properties for relatively different usages. the two broad classes of qualifiers are:

- **Element Refinement.** These qualifiers make the meaning of an element narrower or more specific. A refined element shares the meaning of the unqualified element, but with a more restricted scope. A client that does not understand a specific element refinement term should be able to ignore the qualifier and treat the metadata value as if it were an unqualified (broader) element. The definitions of element refinement terms for qualifiers must be publicly available.
- **Encoding Scheme.** These qualifiers identify schemes that aid in the interpretation of an element value. These schemes include controlled vocabularies and formal notations or parsing rules. A value expressed using an encoding scheme will thus be a token selected from a controlled vocabulary (e.g., a term from a classification system or set of subject headings) or a string formatted in accordance with a formal notation (e.g., "2000-01-01" as the standard expression of a date). If an encoding scheme is not understood by a client or agent, the value may still be useful to a human reader. The definitive description of an encoding scheme for qualifiers must be clearly identified and available for public use.

The DC-NMF profile directly supports the element refinement qualifiers specified in the DCQ specification and maps them to first class properties that are defined in the DC Terms schema. Encoding Schema qualifiers are supported indirectly via the qualified property value. The qualified property value wraps the base property value that contains the unqualified value along with one or more properties that specify the additional information that can be used to further interpret the base value.

Dublin Core Qualifiers also describes a simple algorithm for processing qualified property values where the qualifier properties are not understood by the processor. the algorithm is simply to ignore the qualifiers and treat the base value as a stand-alone item.

- An application that makes use of a qualified property value type SHOULD strive to specify the base value such that it will be meaningful to processors that don't understand the qualifiers that are provided.
- An application SHOULD avoid use of qualified property values.

1.4.5.1 Implicit Data Typing

DC-NMF defines the simple property types using a weakly typed string datatype in order to be compatible with regular DC usage that allows overloading of the base property with the actual type being specified either by context or by qualifiers that provide the encoding scheme.

Despite the fact that DC-NMF doesn't explicitly type the simple types using XML Schema datatyping mechanisms, there is an implicit datatype that is assumed in many cases and which is specified in the prose of the property description. This implicit type is described in the best practices description of the property.

- An DC-NMF aware application SHOULD honor the implicit data specified by in the property definition.

1.4.6 Array Properties

Dublin Core metadata properties are allowed to repeat one or more times. This usage can be problematic in many situations where the intent of the repetition is not clear. For example, a given sequence may be an ordered sequence of properties, an unordered sequence of properties, or a set or exclusive alternatives. There are also many processing environments that do not support the occurrence of more than one property with the same name in a composite property.

DC-NMF defines a specific array interpretation for any property where repetition is allowed. The array is either ordered, unordered or alternative. If the DC-NMF metadata was obtained via a translation step from some other DC representation such as RDF it is the responsibility of the translator to supply the mapping logic to map any repeating property occurrences into an array representation.

Chapter 2: Dublin Core Elements Schema

2.1 Best Practices and Usage

The Dublin Core Element Set (DCES) defines fifteen properties that can be used to describe resources. Some of these properties may not be clearly enough defined to allow a high level of interoperability to occur between loosely coupled participants in a metadata interchange scenario.

DCMI is providing increased clarity on the the contents on encoding of the DCES via the definition of qualifiers that either refine the meaning of the core elements or nail down the details of the encoding of the core elements.

There are some elements of DCES that are in wide usage and provide a high degree of interoperability, not the least because they are intended to be weakly typed and not directly machine processable. These elements are:

- description
- title

In addition, DC-NMF specifies a specific interpretation of some of the DCES that allows them to provide a higher degree of utility using the implicit data typing best practice of DC-NMF. The following properties are implicitly typed in DC-NMF:

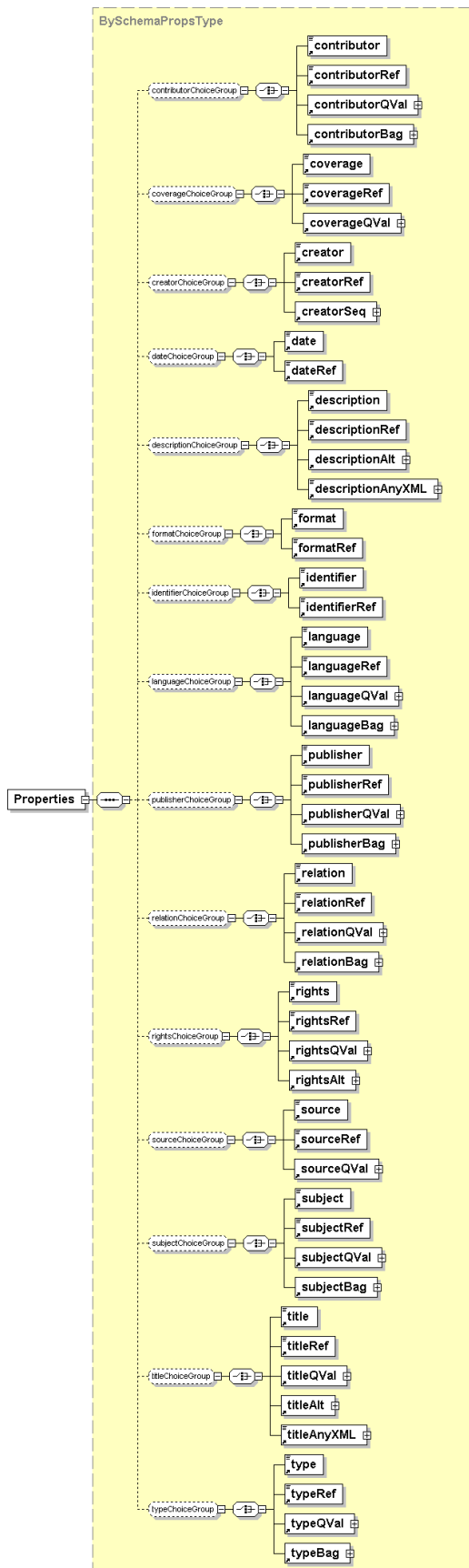
- date
- format
- identifier

2.2 Properties container

The Properties container element contains zero or more properties from the schema. These properties are specified in alphabetical order and can occur zero or once in the element. Each property is specified as a choice group that contains the alternative types that can occur in that position within the Properties container element..

The Properties element is an instance of the `BySchemaPropsType` complexType.

diagram



namespace	http://purl.org/dc/elements/1.1/
type	BySchemaPropsType
children	contributor contributorRef contributorQVal contributorBag coverage coverageRef coverageQVal creator creatorRef creatorSeq date dateRef description descriptionRef descriptionAlt descriptionAnyXML format formatRef identifier identifierRef language languageRef languageQVal languageBag publisher publisherRef publisherQVal publisherBag relation relationRef relationQVal relationBag rights rightsRef rightsQVal rightsAlt source sourceRef sourceQVal subject subjectRef subjectQVal subjectBag title titleRef titleQVal titleAlt titleAnyXML type typeRef typeQVal typeBag
source	<xs:element name="Properties" type="BySchemaPropsType" substitutionGroup="nmf:BySchemaPropsBase"/>

2.3 Property: contributor

Definition

An entity responsible for making contributions to the content of the resource.

Comment

Examples of a Contributor include a person, an organisation, or a service. Typically, the name of a Contributor should be used to indicate the entity.

Practice

This property is weakly typed and MAY NOT be actively supported by DC-NMF processors.

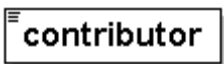
2.3.1 group contributorChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	contributor contributorRef contributorQVal contributorBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="contributorChoiceGroup"> <xs:choice> <xs:element ref="contributor"/> <xs:element ref="contributorRef"/> <xs:element ref="contributorQVal"/> <xs:element ref="contributorBag"/> </xs:choice></pre>

	<code></xs:group></code>
--	--------------------------------

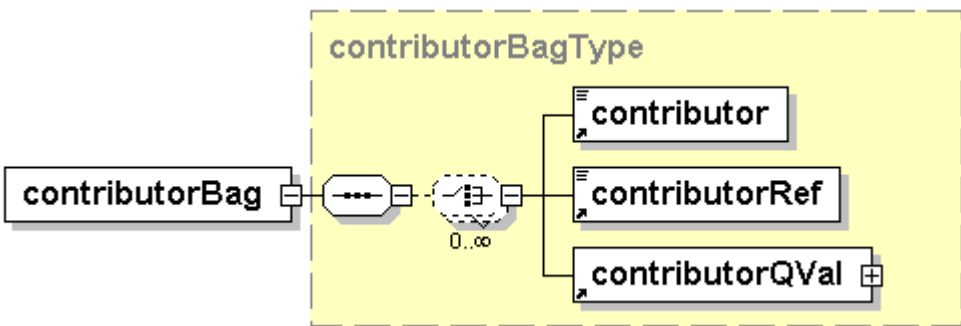
2.3.2 element contributor

contributor is the base property value for the contributor property.

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	contributorType
used by	complexType contributorBagType contributorQValType group contributorChoiceGroup
source	<code><xs:element name="contributor" type="contributorType"/></code>

2.3.3 element contributorBag

In the case where there is more than one contributor, a contributorBag property is used to contain the actual contributor properties. The contributorBag is an unordered container for the contributors.

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	contributorBagType
children	contributor contributorRef contributorQVal
used by	group contributorChoiceGroup
source	<code><xs:element name="contributorBag" type="contributorBagType"/></code>

2.3.4 element contributorQVal

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	contributorQValType
children	contributor contributorRef
used by	complexType contributorBagType group contributorChoiceGroup
source	<code><xs:element name="contributorQVal" type="contributorQValType"/></code>

2.3.5 element contributorRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	contributorRefType
used by	complexType contributorBagType contributorQValType group contributorChoiceGroup
source	<code><xs:element name="contributorRef" type="contributorRefType"/></code>

2.4 *Property: coverage*

Definition

The extent or scope of the content of the resource.

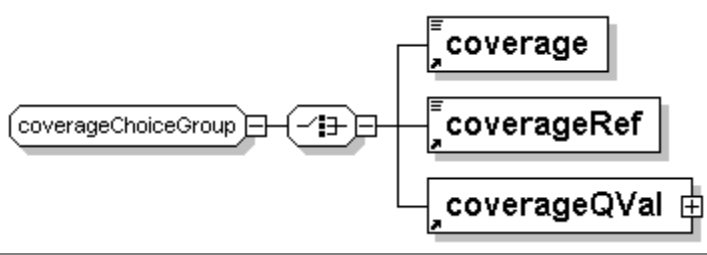
Comment

Coverage will typically include spatial location (a place name or geographic coordinates), temporal period (a period label, date, or date range) or jurisdiction (such as a named administrative entity). Recommended best practice is to select a value from a controlled vocabulary (for example, the Thesaurus of Geographic Names [TGN]) and that, where appropriate, named places or time periods be used in preference to numeric identifiers such as sets of coordinates or date ranges.

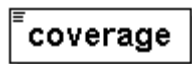
Practice

This property is weakly typed and MAY NOT be actively supported by DC-NMF processors.

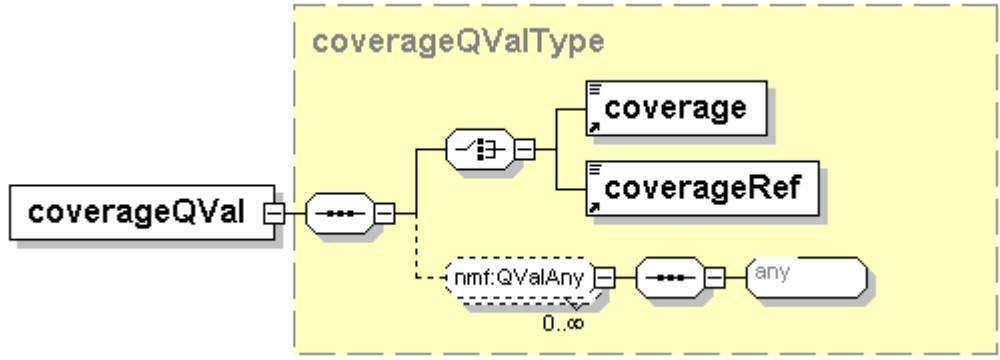
2.4.1 group coverageChoiceGroup

diagram	 <p>The diagram shows a rounded rectangle labeled 'coverageChoiceGroup' connected to a choice symbol (a circle with a vertical bar and a plus sign). This choice symbol is connected to three rectangular boxes representing elements: 'coverage', 'coverageRef', and 'coverageQVal'. Each element box has a small square icon in its top-left corner.</p>
namespace	http://purl.org/dc/elements/1.1/
children	coverage coverageRef coverageQVal
used by	complexType BySchemaPropsType
source	<pre> <xs:group name="coverageChoiceGroup"> <xs:choice> <xs:element ref="coverage"/> <xs:element ref="coverageRef"/> <xs:element ref="coverageQVal"/> </xs:choice> </xs:group> </pre>

2.4.2 element coverage


diagram	 <p>The diagram shows a simple rectangular box labeled 'coverage' with a small square icon in its top-left corner.</p>
namespace	http://purl.org/dc/elements/1.1/
type	coverageType
used by	complexType coverageQValType group coverageChoiceGroup
source	<pre><xs:element name="coverage" type="coverageType"/></pre>

2.4.3 element coverageQVal

diagram	 <p>The diagram shows a rounded rectangle labeled 'coverageQVal' connected to a choice symbol. This choice symbol is connected to two paths. The top path goes through another choice symbol to two boxes labeled 'coverage' and 'coverageRef'. The bottom path goes through a dashed-line rounded rectangle labeled 'nmf:QValAny' with '0..∞' below it, followed by a choice symbol connected to a box labeled 'any'. A yellow dashed box labeled 'coverageQValType' encloses the 'nmf:QValAny' and 'any' components.</p>
---------	--

namespace	http://purl.org/dc/elements/1.1/
type	coverageQValType
children	coverage coverageRef
used by	group coverageChoiceGroup
source	<code><xs:element name="coverageQVal" type="coverageQValType"/></code>

2.4.4 element coverageRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	coverageRefType
used by	complexType coverageQValType group coverageChoiceGroup
source	<code><xs:element name="coverageRef" type="coverageRefType"/></code>

2.5 Property: creator

Definition

An entity primarily responsible for making the content of the resource.

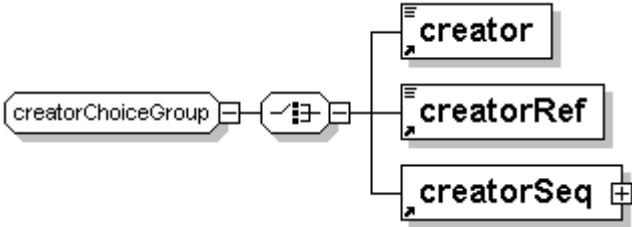
Comment

Examples of a Creator include a person, an organisation, or a service. Typically, the name of a Creator should be used to indicate the entity.

Practice

This property is weakly typed and MAY NOT be actively supported by DC-NMF processors.

2.5.1 group creatorChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	creator creatorRef creatorSeq
used by	complexType BySchemaPropsType
source	<code><xs:group name="creatorChoiceGroup"></code>

	<pre> <xs:choice> <xs:element ref="creator"/> <xs:element ref="creatorRef"/> <xs:element ref="creatorSeq"/> </xs:choice> </xs:group> </pre>
--	---

2.5.2 element creator

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	creatorType
used by	complexType creatorSeqType group creatorChoiceGroup
source	<xs:element name="creator" type="creatorType"/>

2.5.3 element creatorRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	creatorRefType
used by	complexType creatorSeqType group creatorChoiceGroup
source	<xs:element name="creatorRef" type="creatorRefType"/>

2.5.4 element creatorSeq

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	creatorSeqType
children	creator creatorRef
used by	group creatorChoiceGroup
source	<xs:element name="creatorSeq" type="creatorSeqType"/>

2.6 Property: date

Definition

A date associated with an event in the life cycle of the resource.

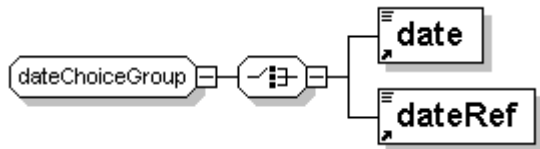
Comment

Typically, Date will be associated with the creation or availability of the resource. Recommended best practice for encoding the date value is defined in a profile of ISO 8601 [W3CDTF] and follows the YYYY-MM-DD format.


Practice

This property is interpreted as conforming to the XML Schema datetime datatype. It SHOULD NOT be used directly in metadata. Instead one of its derived types defined in the terms schema should be used.


2.6.1 group dateChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	date dateRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="dateChoiceGroup"> <xs:choice> <xs:element ref="date"/> <xs:element ref="dateRef"/> </xs:choice> </xs:group></pre>

2.6.2 element date

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	dateType
used by	group dateChoiceGroup
source	<pre><xs:element name="date" type="dateType"/></pre>

2.6.3 element dateRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	dateRefType
used by	group dateChoiceGroup
source	<code><xs:element name="dateRef" type="dateRefType"/></code>

2.7 Property: description

Definition

An account of the content of the resource.

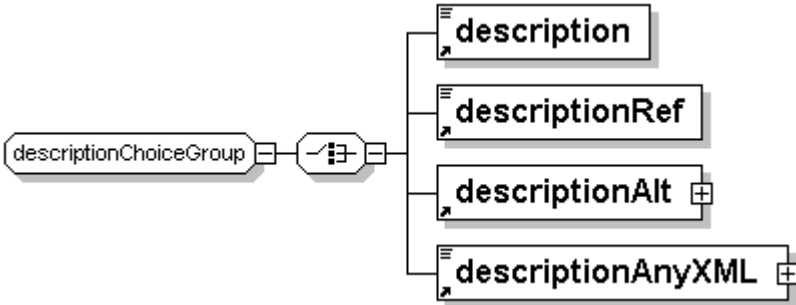
Comment

Practice

This property is used for longer human readable accounts of the resource. A short concise human readable label should be specified using the title property.

If well-formed XML is used in the description, the descriptionAnyXML variant of the property should be used.

2.7.1 group descriptionChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	description descriptionRef descriptionAlt descriptionAnyXML
used by	complexType BySchemaPropsType
source	<pre><xs:group name="descriptionChoiceGroup"> <xs:choice> <xs:element ref="description"/> <xs:element ref="descriptionRef"/> </xs:choice> </xs:group></pre>

	<pre> <xs:element ref="descriptionAlt"/> <xs:element ref="descriptionAnyXML"/> </xs:choice> </xs:group> </pre>
--	--

2.7.2 element description

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	descriptionType
used by	complexType descriptionAltType group descriptionChoiceGroup
source	<code><xs:element name="description" type="descriptionType"/></code>

2.7.3 element descriptionAlt


diagram	
namespace	http://purl.org/dc/elements/1.1/
type	descriptionAltType
children	description descriptionRef descriptionAnyXML
used by	group descriptionChoiceGroup
source	<code><xs:element name="descriptionAlt" type="descriptionAltType"/></code>

2.7.4 element descriptionAnyXML

diagram	
namespace	http://purl.org/dc/elements/1.1/

type	descriptionAnyXMLType
used by	complexType descriptionAltType group descriptionChoiceGroup
source	<code><xs:element name="descriptionAnyXML" type="descriptionAnyXMLType"/></code>

2.7.5 element descriptionRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	descriptionRefType
used by	complexType descriptionAltType group descriptionChoiceGroup
source	<code><xs:element name="descriptionRef" type="descriptionRefType"/></code>

2.8 Property: format

Definition

The physical or digital manifestation of the resource.

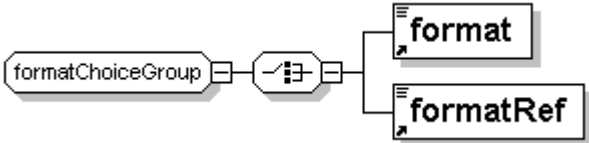
Comment

Typically, 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. Recommended best practice is to select a value from a controlled vocabulary (for example, the list of Internet Media Types [MIME] defining computer media formats).

Practice


This property is interpreted as using the [MIME] controlled. It MAY be used to describe the content-type of the resource.

2.8.1 group formatChoiceGroup


diagram	
namespace	http://purl.org/dc/elements/1.1/
children	format formatRef
used by	complexType BySchemaPropsType
source	<code><xs:group name="formatChoiceGroup"> <xs:choice> <xs:element ref="format"/></code>

	<pre><xs:element ref="formatRef"/> </xs:choice> </xs:group></pre>
--	---

2.8.2 element format

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	formatType
used by	group formatChoiceGroup
source	<pre><xs:element name="format" type="formatType"/></pre>

2.8.3 element formatRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	formatRefType
used by	group formatChoiceGroup
source	<pre><xs:element name="formatRef" type="formatRefType"/></pre>

2.9 *Property: identifier*

Definition

An unambiguous reference to the resource within a given context.

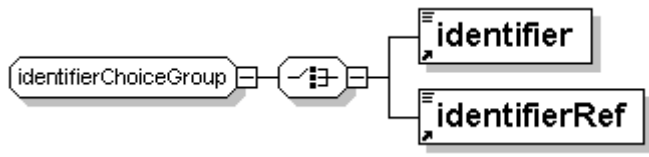
Comment

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).


Practice

This property is implicitly typed to be conform to the XML Schema xs:anyURI datatype.


2.9.1 group identifierChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	identifier identifierRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="identifierChoiceGroup"> <xs:choice> <xs:element ref="identifier"/> <xs:element ref="identifierRef"/> </xs:choice> </xs:group></pre>

2.9.2 element identifier

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	identifierType
used by	group identifierChoiceGroup
source	<pre><xs:element name="identifier" type="identifierType"/></pre>

2.9.3 element identifierRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	identifierRefType
used by	group identifierChoiceGroup
source	<pre><xs:element name="identifierRef" type="identifierRefType"/></pre>

2.10 Property: 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 1766 [RFC1766] which includes a two-letter Language Code (taken from the ISO 639 standard [ISO639]), followed optionally, by a two-letter Country Code (taken from the ISO 3166 standard [ISO3166]). For example, 'en' for English, 'fr' for French, or 'en-uk' for English used in the United Kingdom.

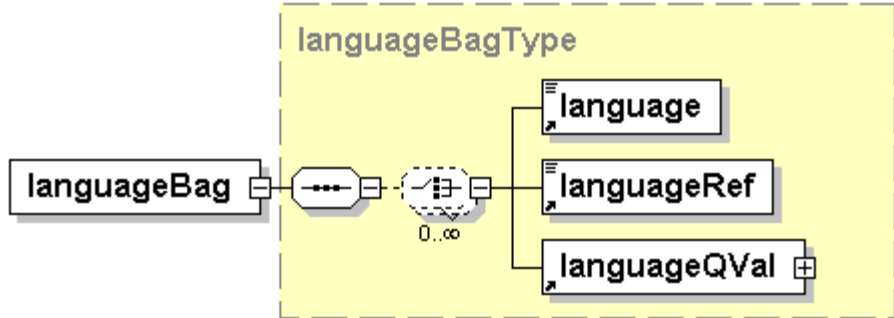
2.10.1 group languageChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	language languageRef languageQVal languageBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="languageChoiceGroup"> <xs:choice> <xs:element ref="language"/> <xs:element ref="languageRef"/> <xs:element ref="languageQVal"/> <xs:element ref="languageBag"/> </xs:choice> </xs:group></pre>

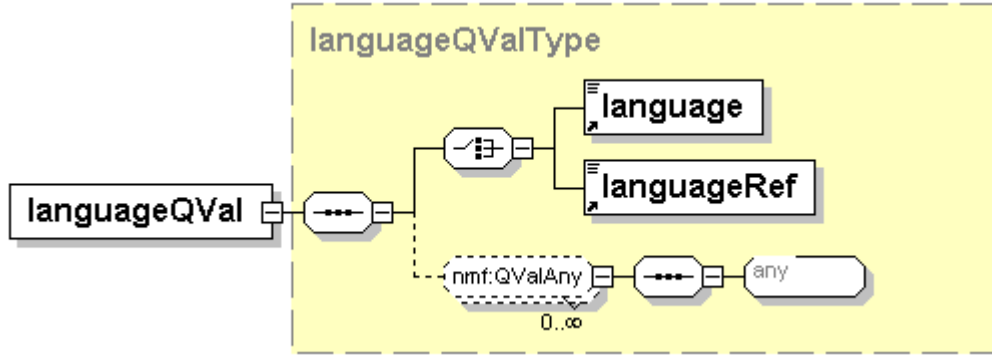
2.10.2 element language

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	languageType
used by	complexTypes languageBagType languageQValType group languageChoiceGroup
source	<pre><xs:element name="language" type="languageType"/></pre>


2.10.3 element languageBag

diagram	 <p>The diagram shows the structure of the <code>languageBagType</code>. It starts with a <code>languageBag</code> element, which is connected to a sequence container (represented by a rounded rectangle with three dots). This sequence container is followed by a choice container (represented by a circle with a vertical line and a plus sign) that has a cardinality of <code>0..∞</code>. The choice container contains three elements: <code>language</code>, <code>languageRef</code>, and <code>languageQVal</code>.</p>
namespace	http://purl.org/dc/elements/1.1/
type	languageBagType
children	language languageRef languageQVal
used by	group languageChoiceGroup
source	<code><xs:element name="languageBag" type="languageBagType"/></code>

2.10.4 element languageQVal

diagram	 <p>The diagram shows the structure of the <code>languageQValType</code>. It starts with a <code>languageQVal</code> element, which is connected to a sequence container (represented by a rounded rectangle with three dots). This sequence container is followed by a choice container (represented by a circle with a vertical line and a plus sign). The choice container has two branches: one leading to a choice container (circle with a vertical line and a plus sign) containing <code>language</code> and <code>languageRef</code>, and another leading to a sequence container (rounded rectangle with three dots) containing <code>nmf:QValAny</code> (indicated by a dashed line) and <code>any</code>. The <code>nmf:QValAny</code> element has a cardinality of <code>0..∞</code>.</p>
namespace	http://purl.org/dc/elements/1.1/
type	languageQValType
children	language languageRef
used by	complexType languageBagType group languageChoiceGroup
source	<code><xs:element name="languageQVal" type="languageQValType"/></code>

2.10.5 element languageRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	languageRefType
used by	complexType group languageBagType languageQValType languageChoiceGroup
source	<code><xs:element name="languageRef" type="languageRefType"/></code>

2.11 Property: publisher

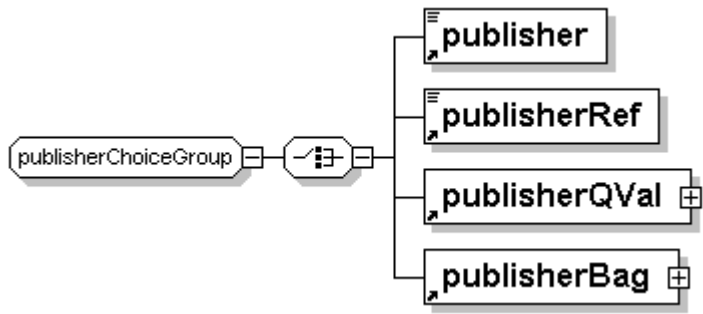
Definition

An entity responsible for making the resource available.


Comment

Examples of a Publisher include a person, an organisation, or a service. Typically, the name of a Publisher should be used to indicate the entity.

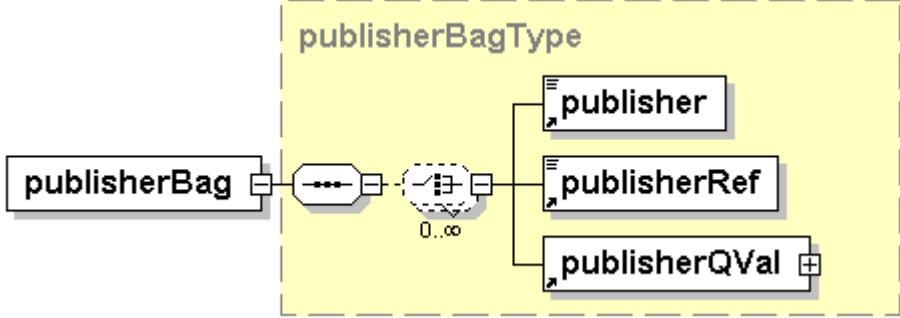
2.11.1 group publisherChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	publisher publisherRef publisherQVal publisherBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="publisherChoiceGroup"> <xs:choice> <xs:element ref="publisher"/> <xs:element ref="publisherRef"/> <xs:element ref="publisherQVal"/> <xs:element ref="publisherBag"/> </xs:choice> </xs:group></pre>

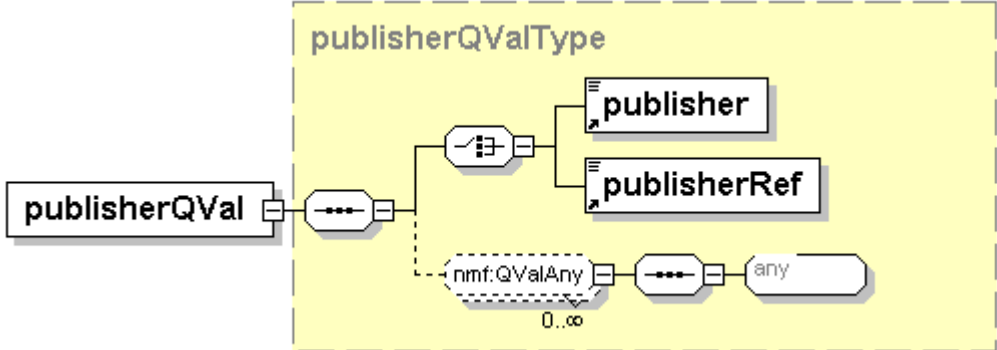
2.11.2 element publisher

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	publisherType
used by	complexTypes publisherBagType publisherQValType group publisherChoiceGroup
source	<code><xs:element name="publisher" type="publisherType"/></code>

2.11.3 element publisherBag


diagram	
namespace	http://purl.org/dc/elements/1.1/
type	publisherBagType
children	publisher publisherRef publisherQVal
used by	group publisherChoiceGroup
source	<code><xs:element name="publisherBag" type="publisherBagType"/></code>

2.11.4 element publisherQVal

diagram	
---------	--

namespace	http://purl.org/dc/elements/1.1/
type	publisherQValType
children	publisher publisherRef
used by	complexType publisherBagType group publisherChoiceGroup
source	<xs:element name="publisherQVal" type="publisherQValType"/>

2.11.5 element publisherRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	publisherRefType
used by	complexTypes publisherBagType publisherQValType group publisherChoiceGroup
source	<xs:element name="publisherRef" type="publisherRefType"/>

2.12 Property: relation

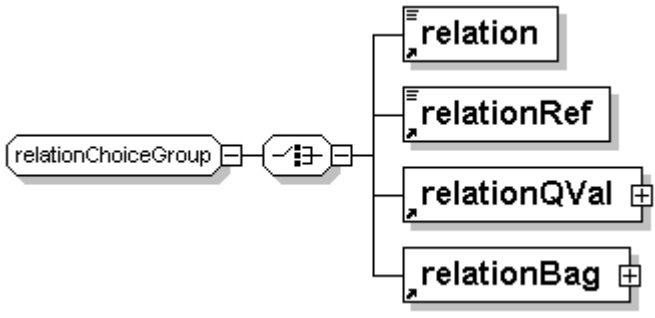
Definition

A reference to a related resource.

Comment

Recommended best practice is to reference the resource by means of a string or number conforming to a formal identification system.

2.12.1 group relationChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	relation relationRef relationQVal relationBag
used by	complexType BySchemaPropsType

source	<pre> <xs:group name="relationChoiceGroup"> <xs:choice> <xs:element ref="relation"/> <xs:element ref="relationRef"/> <xs:element ref="relationQVal"/> <xs:element ref="relationBag"/> </xs:choice> </xs:group> </pre>
--------	---

2.12.2 element relation

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	relationType
used by	complexTypes relationBagType relationQValType group relationChoiceGroup
source	<code><xs:element name="relation" type="relationType"/></code>

2.12.3 element relationBag

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	relationBagType
children	relation relationRef relationQVal
used by	group relationChoiceGroup
source	<code><xs:element name="relationBag" type="relationBagType"/></code>

2.12.4 element relationQVal

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	relationQValType
children	relation relationRef
used by	complexType relationBagType group relationChoiceGroup
source	<code><xs:element name="relationQVal" type="relationQValType"/></code>

2.12.5 element relationRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	relationRefType
used by	complexTypes relationBagType relationQValType group relationChoiceGroup
source	<code><xs:element name="relationRef" type="relationRefType"/></code>

2.13 Property: rights

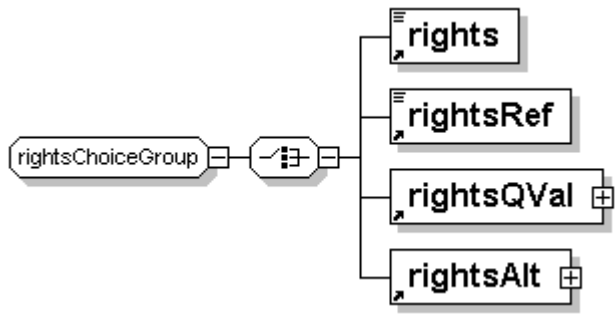
Definition

Information about rights held in and over the resource.


Comment

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.

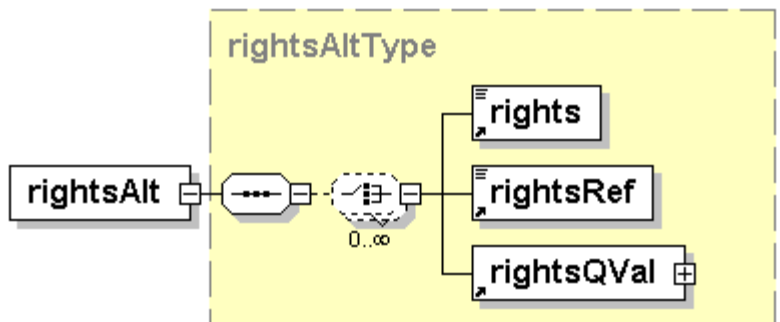
2.13.1 group rightsChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	rights rightsRef rightsQVal rightsAlt
used by	complexType BySchemaPropsType
source	<pre><xs:group name="rightsChoiceGroup"> <xs:choice> <xs:element ref="rights"/> <xs:element ref="rightsRef"/> <xs:element ref="rightsQVal"/> <xs:element ref="rightsAlt"/> </xs:choice> </xs:group></pre>

2.13.2 element rights

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	rightsType
used by	complexType rightsAltType rightsQValType group rightsChoiceGroup
source	<pre><xs:element name="rights" type="rightsType"/></pre>

2.13.3 element rightsAlt

diagram	
namespace	http://purl.org/dc/elements/1.1/

type	rightsAltType
children	rights rightsRef rightsQVal
used by	group rightsChoiceGroup
source	<code><xs:element name="rightsAlt" type="rightsAltType"/></code>

2.13.4 element rightsQVal

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	rightsQValType
children	rights rightsRef
used by	complexType rightsAltType group rightsChoiceGroup
source	<code><xs:element name="rightsQVal" type="rightsQValType"/></code>

2.13.5 element rightsRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	rightsRefType
used by	complexTypes rightsAltType rightsQValType group rightsChoiceGroup
source	<code><xs:element name="rightsRef" type="rightsRefType"/></code>

2.14 Property: source

Definition

A Reference to a resource from which the present resource is derived.

Comment

The present resource may be derived from the Source resource in whole or in part. Recommended best practice is to reference the resource by means of a string or number conforming to a formal identification system.

2.14.1 group sourceChoiceGroup

diagram	
namespace	http://purl.org/dc/elements/1.1/
children	source sourceRef sourceQVal
used by	complexType BySchemaPropsType
source	<pre><xs:group name="sourceChoiceGroup"> <xs:choice> <xs:element ref="source"/> <xs:element ref="sourceRef"/> <xs:element ref="sourceQVal"/> </xs:choice> </xs:group></pre>

2.14.2 element source

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	sourceType
used by	complexType sourceQValType group sourceChoiceGroup
source	<pre><xs:element name="source" type="sourceType"/></pre>

2.14.3 element sourceQVal

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	sourceQValType
children	source sourceRef
used by	group sourceChoiceGroup
source	<code><xs:element name="sourceQVal" type="sourceQValType"/></code>

2.14.4 element sourceRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	sourceRefType
used by	complexType sourceQValType group sourceChoiceGroup
source	<code><xs:element name="sourceRef" type="sourceRefType"/></code>

2.15 Property: subject

Definition

The topic of the content of the resource.

Comment

Typically, a Subject will be expressed as keywords, key phrases or classification codes that describe a topic of the resource. Recommended best practice is to select a value from a controlled vocabulary or formal classification scheme. Multiple values can be separated by semicolon “;” and a semicolon in the value can be escaped using the backslash “\”.

2.15.1 group subjectChoiceGroup

diagram	<p>The diagram shows the structure of the <code>subjectChoiceGroup</code> element. It is a container element (rectangle with a small square on the left) that contains a choice element (circle with a vertical line and a small square on the left). This choice element contains four child elements: <code>subject</code>, <code>subjectRef</code>, <code>subjectQVal</code>, and <code>subjectBag</code>. Each child element is represented by a rectangle with a small square on the left and a small square on the right, indicating that each child is optional and can occur once.</p>
namespace	http://purl.org/dc/elements/1.1/
children	subject subjectRef subjectQVal subjectBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="subjectChoiceGroup"> <xs:choice> <xs:element ref="subject"/> <xs:element ref="subjectRef"/> <xs:element ref="subjectQVal"/> <xs:element ref="subjectBag"/> </xs:choice> </xs:group></pre>

2.15.2 element subject

diagram	<p>The diagram shows the structure of the <code>subject</code> element, which is a simple element represented by a rectangle with a small square on the left.</p>
namespace	http://purl.org/dc/elements/1.1/
type	subjectType
used by	complexTypes subjectBagType subjectQValType group subjectChoiceGroup
source	<pre><xs:element name="subject" type="subjectType"/></pre>

2.15.3 element subjectBag

diagram	<p>The diagram shows the structure of the <code>subjectBag</code> element. It is a container element (rectangle with a small square on the left) that contains a choice element (circle with a vertical line and a small square on the left). This choice element contains three child elements: <code>subject</code>, <code>subjectRef</code>, and <code>subjectQVal</code>. Each child element is represented by a rectangle with a small square on the left and a small square on the right, indicating that each child is optional and can occur once. The choice element is enclosed in a dashed yellow box labeled <code>subjectBagType</code>. The choice element has a cardinality of <code>0..∞</code>.</p>
---------	--

namespace	http://purl.org/dc/elements/1.1/
type	subjectBagType
children	subject subjectRef subjectQVal
used by	group subjectChoiceGroup
source	<code><xs:element name="subjectBag" type="subjectBagType"/></code>

2.15.4 element [subjectQVal](#)

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	subjectQValType
children	subject subjectRef
used by	complexType subjectBagType group subjectChoiceGroup
source	<code><xs:element name="subjectQVal" type="subjectQValType"/></code>

2.15.5 element [subjectRef](#)

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	subjectRefType
used by	complexTypes subjectBagType subjectQValType group subjectChoiceGroup
source	<code><xs:element name="subjectRef" type="subjectRefType"/></code>

2.16 Property: title

Definition

A name given to the resource.

Comment

Typically, a Title will be a name by which the resource is formally known.

Practice

This property is used for a short one line label that can be displayed in an index list of resources such as a resource directory browser.

2.16.1 group titleChoiceGroup

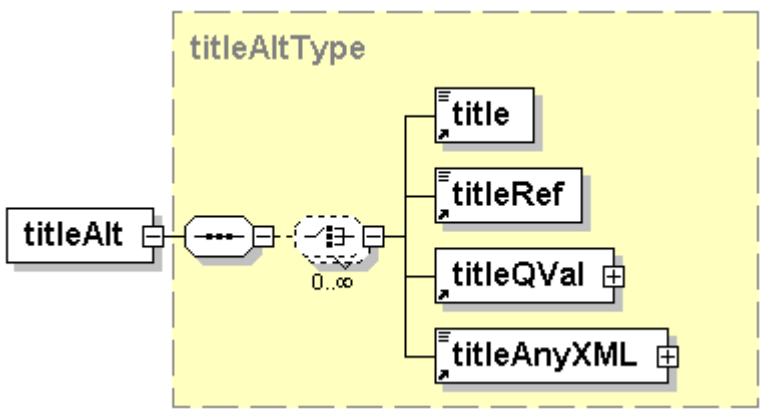
diagram	
namespace	http://purl.org/dc/elements/1.1/
children	title titleRef titleQVal titleAlt titleAnyXML
used by	complexType BySchemaPropsType
source	<pre><xs:group name="titleChoiceGroup"> <xs:choice> <xs:element ref="title"/> <xs:element ref="titleRef"/> <xs:element ref="titleQVal"/> <xs:element ref="titleAlt"/> <xs:element ref="titleAnyXML"/> </xs:choice> </xs:group></pre>

2.16.2 element title

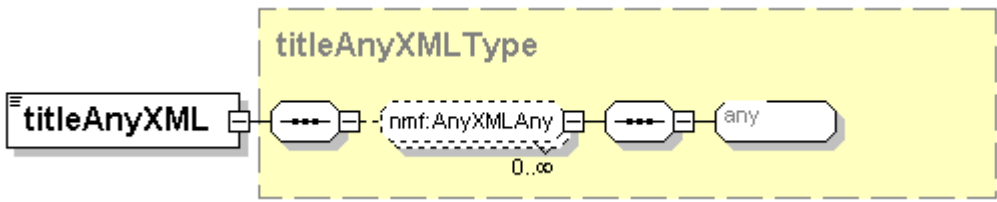
diagram	
namespace	http://purl.org/dc/elements/1.1/

type	titleType
used by	complexType group titleAltType titleQValType titleChoiceGroup
source	<code><xs:element name="title" type="titleType"/></code>

2.16.3 element titleAlt

diagram	 <p>The diagram shows the structure of the <code>titleAltType</code>. It is a sequence of elements: <code>titleAlt</code>, a sequence container (represented by a box with three dots), a choice container (represented by a dashed box with a plus sign and <code>0..∞</code> below it), and a sequence of four elements: <code>title</code>, <code>titleRef</code>, <code>titleQVal</code>, and <code>titleAnyXML</code>. The <code>titleAlt</code> element is shown as a box with a plus sign. The choice container is shown as a dashed box with a plus sign and <code>0..∞</code> below it. The <code>title</code>, <code>titleRef</code>, <code>titleQVal</code>, and <code>titleAnyXML</code> elements are shown as boxes with plus signs.</p>
namespace	<code>http://purl.org/dc/elements/1.1/</code>
type	titleAltType
children	title titleRef titleQVal titleAnyXML
used by	group titleChoiceGroup
source	<code><xs:element name="titleAlt" type="titleAltType"/></code>

2.16.4 element titleAnyXML

diagram	 <p>The diagram shows the structure of the <code>titleAnyXMLType</code>. It is a sequence of elements: <code>titleAnyXML</code>, a sequence container (represented by a box with three dots), a choice container (represented by a dashed box with a plus sign and <code>0..∞</code> below it), a sequence container (represented by a box with three dots), and an <code>any</code> element. The <code>titleAnyXML</code> element is shown as a box with a plus sign. The choice container is shown as a dashed box with a plus sign and <code>0..∞</code> below it. The <code>any</code> element is shown as a box.</p>
namespace	<code>http://purl.org/dc/elements/1.1/</code>
type	titleAnyXMLType
used by	complexType titleAltType group titleChoiceGroup
source	<code><xs:element name="titleAnyXML" type="titleAnyXMLType"/></code>

2.16.5 element titleQVal

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	titleQValType
children	title titleRef
used by	complexType titleAltType group titleChoiceGroup
source	<code><xs:element name="titleQVal" type="titleQValType"/></code>

2.16.6 element titleRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	titleRefType
used by	complexTypes titleAltType titleQValType group titleChoiceGroup
source	<code><xs:element name="titleRef" type="titleRefType"/></code>

2.17 Property: type

Definition

The nature or genre of the content of the resource.

Comment

2.17.1 **group** typeChoiceGroup

diagram	<p>The diagram shows a 'typeChoiceGroup' element connected to a choice element (represented by a circle with a vertical bar and a plus sign). This choice element branches into four child elements: 'type', 'typeRef', 'typeQVal', and 'typeBag'. Each child element has a small square icon with a plus sign in the top-left corner, indicating it is a complex type.</p>
namespace	http://purl.org/dc/elements/1.1/
children	type typeRef typeQVal typeBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="typeChoiceGroup"> <xs:choice> <xs:element ref="type"/> <xs:element ref="typeRef"/> <xs:element ref="typeQVal"/> <xs:element ref="typeBag"/> </xs:choice> </xs:group></pre>

2.17.2 **element** type

diagram	<p>The diagram shows a single 'type' element represented by a rectangle with a small square icon in the top-left corner.</p>
namespace	http://purl.org/dc/elements/1.1/
type	typeType
used by	complexTypees typeBagType typeQValType group typeChoiceGroup
source	<pre><xs:element name="type" type="typeType"/></pre>

2.17.3 **element** typeBag

diagram	<p>The diagram shows a 'typeBag' element connected to a choice element (represented by a circle with a vertical bar and a plus sign). This choice element branches into three child elements: 'type', 'typeRef', and 'typeQVal'. Each child element has a small square icon with a plus sign in the top-left corner. The choice element is enclosed in a dashed yellow box labeled 'typeBagType'. Below the choice element is the cardinality '0..∞'.</p>
---------	---

namespace	http://purl.org/dc/elements/1.1/
type	typeBagType
children	type typeRef typeQVal
used by	group typeChoiceGroup
source	<code><xs:element name="typeBag" type="typeBagType"/></code>

2.17.4 element typeQVal

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	typeQValType
children	type typeRef
used by	complexType typeBagType group typeChoiceGroup
source	<code><xs:element name="typeQVal" type="typeQValType"/></code>

2.17.5 element typeRef

diagram	
namespace	http://purl.org/dc/elements/1.1/
type	typeRefType
used by	complexTypes typeBagType typeQValType group typeChoiceGroup
source	<code><xs:element name="typeRef" type="typeRefType"/></code>

Chapter 3: Dublin Core Terms Schema

3.1 Best Practices and Usage

The Dublin Core Terms schema [DCQ] defines a large set of qualifiers for the DCES. These qualifiers are split into qualifiers that refine the meaning of the DCES properties and qualifiers that provide context for how to interpret the encoding of the DCES properties or DCTerms refined properties.

DC-NMF doesn't directly support the encoding qualifiers other than to allow metadata that uses them to be weakly encoded in DC-NMF using the qualified property type. Instead, DC-NMF explicitly defines the element refinements of the Dublin Core Qualifiers specification as first class properties and uses implicit data typing where appropriate to specify the encoding.

Date Properties

The refinements of the date property in DCES are of general utility for many applications since they provide a useful set of temporal properties. Specifically, DC-NMF applications are encouraged to use the following properties for increased interoperability. Note that all these properties derive from the `dc:date` property which is implicitly typed to conform to the XML schema `datetime` datatype.

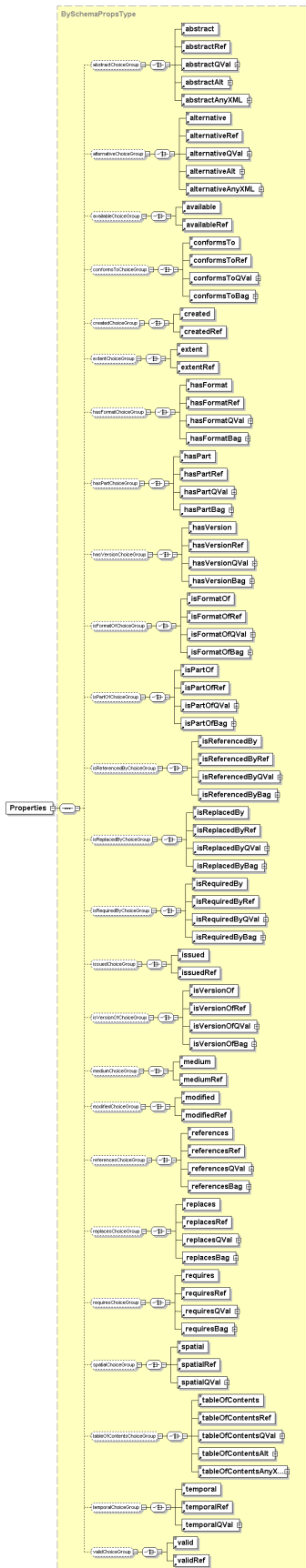
- `created`
- `modified`

3.2 Properties Container

The Properties container element contains zero or more properties from the schema. These properties are specified in alphabetical order and can occur zero or once in the element. Each property is specified as a choice group that contains the alternative types that can occur in that position within the Properties container element.

The Properties element is an instance of the `BySchemaPropsType` complexType.

diagram



namespace	http://purl.org/dc/terms/
type	BySchemaPropsType
children	abstract abstractRef abstractQVal abstractAlt abstractAnyXML alternative alternativeRef alternativeQVal alternativeAlt alternativeAnyXML available availableRef conformsTo conformsToRef conformsToQVal conformsToBag created createdRef extent extentRef hasFormat hasFormatRef hasFormatQVal hasFormatBag hasPart hasPartRef hasPartQVal hasPartBag hasVersion hasVersionRef hasVersionQVal hasVersionBag isFormatOf isFormatOfRef isFormatOfQVal isFormatOfBag isPartOf isPartOfRef isPartOfQVal isPartOfBag isReferencedBy isReferencedByRef isReferencedByQVal isReferencedByBag isReplacedBy isReplacedByRef isReplacedByQVal isReplacedByBag issued issuedRef isVersionOf isVersionOfRef isVersionOfQVal isVersionOfBag medium mediumRef modified modifiedRef references referencesRef referencesQVal referencesBag replaces replacesRef replacesQVal replacesBag requires requiresRef requiresQVal requiresBag spatial spatialRef spatialQVal tableOfContents tableOfContentsRef tableOfContentsQVal tableOfContentsAlt tableOfContentsAnyXML temporal temporalRef temporalQVal valid validRef
source	<xs:element name="Properties" type="BySchemaPropsType" substitutionGroup="nmf:BySchemaPropsBase"/>

3.3 Property: abstract

Refines

description

Definition

A summary of the content of the resource.

Comment

3.3.1 group abstractChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	abstract abstractRef abstractQVal abstractAlt abstractAnyXML
used by	complexType BySchemaPropsType

source	<pre> <xs:group name="abstractChoiceGroup"> <xs:choice> <xs:element ref="abstract"/> <xs:element ref="abstractRef"/> <xs:element ref="abstractQVal"/> <xs:element ref="abstractAlt"/> <xs:element ref="abstractAnyXML"/> </xs:choice> </xs:group> </pre>
--------	--

3.3.2 element abstract

diagram	
namespace	http://purl.org/dc/terms/
type	abstractType
used by	complexType abstractAltType abstractQValType group abstractChoiceGroup
source	<code><xs:element name="abstract" type="abstractType"/></code>

3.3.3 element abstractAlt

diagram	
namespace	http://purl.org/dc/terms/
type	abstractAltType
children	abstract
used by	group abstractChoiceGroup
source	<code><xs:element name="abstractAlt" type="abstractAltType"/></code>

3.3.4 element abstractAnyXML

diagram	
namespace	http://purl.org/dc/terms/
type	abstractAnyXMLType
used by	group abstractChoiceGroup
source	<code><xs:element name="abstractAnyXML" type="abstractAnyXMLType"/></code>

3.3.5 element abstractQVal

diagram	
namespace	http://purl.org/dc/terms/
type	abstractQValType
children	abstract
used by	group abstractChoiceGroup
source	<code><xs:element name="abstractQVal" type="abstractQValType"/></code>

3.3.6 element abstractRef

diagram	
namespace	http://purl.org/dc/terms/
type	abstractRefType
used by	group abstractChoiceGroup
source	<code><xs:element name="abstractRef" type="abstractRefType"/></code>

3.4 Property: *alternative*

Definition

Any form of the title used as a substitute or alternative to the formal title of the resource.

Comment

3.4.1 group alternativeChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	alternative alternativeRef alternativeQVal alternativeAlt alternativeAnyXML
used by	complexType BySchemaPropsType
source	<pre><xs:group name="alternativeChoiceGroup"> <xs:choice> <xs:element ref="alternative"/> <xs:element ref="alternativeRef"/> <xs:element ref="alternativeQVal"/> <xs:element ref="alternativeAlt"/> <xs:element ref="alternativeAnyXML"/> </xs:choice> </xs:group></pre>

3.4.2 element alternative

diagram	
namespace	http://purl.org/dc/terms/
type	alternativeType
used by	complexType alternativeAltType alternativeQValType group alternativeChoiceGroup
source	<pre><xs:element name="alternative" type="alternativeType"/></pre>

3.4.3 element alternativeAlt

diagram	<p>The diagram shows the alternativeAlt element connected to a sequence of elements. The sequence starts with a sequence container (rectangle with four dots), followed by an alternative container (dashed circle with a vertical bar and a slash), and then an alternative element. The alternative container is labeled with 0..∞ below it. The entire sequence is enclosed in a dashed yellow box labeled alternativeAltType.</p>
namespace	http://purl.org/dc/terms/
type	alternativeAltType
children	alternative
used by	group alternativeChoiceGroup
source	<code><xs:element name="alternativeAlt" type="alternativeAltType"/></code>

3.4.4 element alternativeAnyXML


diagram	<p>The diagram shows the alternativeAnyXML element connected to a sequence of elements. The sequence starts with a sequence container (rectangle with four dots), followed by an nmf:AnyXMLAny element (dashed circle), another sequence container (rectangle with four dots), and finally an any element (rounded rectangle). The nmf:AnyXMLAny element is labeled with 0..∞ below it. The entire sequence is enclosed in a dashed yellow box labeled alternativeAnyXMLType.</p>
namespace	http://purl.org/dc/terms/
type	alternativeAnyXMLType
used by	group alternativeChoiceGroup
source	<code><xs:element name="alternativeAnyXML" type="alternativeAnyXMLType"/></code>

3.4.5 element alternativeQVal

diagram	<p>The diagram shows the alternativeQVal element connected to a sequence of elements. The sequence starts with a sequence container (rectangle with four dots), followed by an alternative container (dashed circle with a vertical bar and a slash), and then an alternative element. The alternative container is labeled with 0..∞ below it. The entire sequence is enclosed in a dashed yellow box labeled alternativeQValType.</p>
namespace	http://purl.org/dc/terms/

type	alternativeQValType
children	alternative
used by	group alternativeChoiceGroup
source	<code><xs:element name="alternativeQVal" type="alternativeQValType"/></code>

3.4.6 element `alternativeRef`

diagram	
namespace	<code>http://purl.org/dc/terms/</code>
type	alternativeRefType
used by	group alternativeChoiceGroup
source	<code><xs:element name="alternativeRef" type="alternativeRefType"/></code>

3.5 Property: *available*

Refines

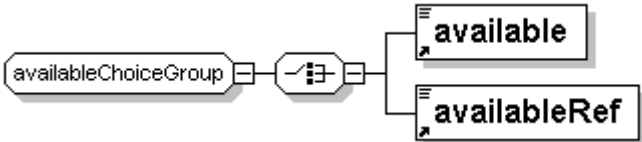
dc:date

Definition


Date (often a range) that the resource will become or did become available.

Comment

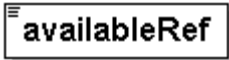
3.5.1 group `availableChoiceGroup`

diagram	
namespace	<code>http://purl.org/dc/terms/</code>
children	available availableRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="availableChoiceGroup"> <xs:choice> <xs:element ref="available"/> <xs:element ref="availableRef"/> </xs:choice> </xs:group></pre>

3.5.2 element available

diagram	
namespace	http://purl.org/dc/terms/
type	availableType
used by	group availableChoiceGroup
source	<code><xs:element name="available" type="availableType"/></code>

3.5.3 element availableRef

diagram	
namespace	http://purl.org/dc/terms/
type	availableRefType
used by	group availableChoiceGroup
source	<code><xs:element name="availableRef" type="availableRefType"/></code>

3.6 Property: conformsTo

Definition

An entity responsible for making contributions to the content of the resource.

Comment

Examples of a Contributor include a person, an organisation, or a service. Typically, the name of a Contributor should be used to indicate the entity.

3.6.1 group conformsToChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	conformsTo conformsToRef conformsToQVal conformsToBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="conformsToChoiceGroup"> <xs:choice> <xs:element ref="conformsTo"/> <xs:element ref="conformsToRef"/> <xs:element ref="conformsToQVal"/> <xs:element ref="conformsToBag"/> </xs:choice> </xs:group></pre>

3.6.2 element conformsTo

diagram	
namespace	http://purl.org/dc/terms/
type	conformsToType
used by	complexTypes conformsToBagType conformsToQValType group conformsToChoiceGroup
source	<pre><xs:element name="conformsTo" type="conformsToType"/></pre>

3.6.3 element conformsToBag

diagram	
namespace	http://purl.org/dc/terms/
type	conformsToBagType
children	conformsTo

used by	group conformsToChoiceGroup
source	<code><xs:element name="conformsToBag" type="conformsToBagType"/></code>

3.6.4 element conformsToQVal

diagram	
namespace	http://purl.org/dc/terms/
type	conformsToQValType
children	conformsTo
used by	group conformsToChoiceGroup
source	<code><xs:element name="conformsToQVal" type="conformsToQValType"/></code>

3.6.5 element conformsToRef

diagram	
namespace	http://purl.org/dc/terms/
type	conformsToRefType
used by	group conformsToChoiceGroup
source	<code><xs:element name="conformsToRef" type="conformsToRefType"/></code>

3.7 Property: created

Refines

date

Definition

Date of creation of the resource.

Practice

This property is specified using the xs:datetime syntax which conforms to ISO 8601.

3.7.1 group createdChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	created createdRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="createdChoiceGroup"> <xs:choice> <xs:element ref="created"/> <xs:element ref="createdRef"/> </xs:choice> </xs:group></pre>

3.7.2 element created

diagram	
namespace	http://purl.org/dc/terms/
type	createdType
used by	group createdChoiceGroup
source	<pre><xs:element name="created" type="createdType"/></pre>

3.7.3 element createdRef

diagram	
namespace	http://purl.org/dc/terms/
type	createdRefType
used by	group createdChoiceGroup
source	<pre><xs:element name="createdRef" type="createdRefType"/></pre>

3.8 Property: extent

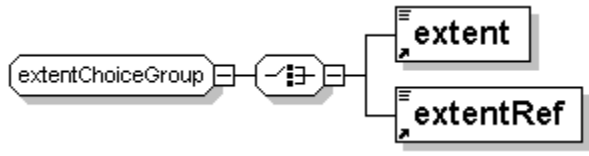
Refines

dc:format


Definition

The size or duration of the resource.


Comment**3.8.1 group extentChoiceGroup**

diagram	
namespace	http://purl.org/dc/terms/
children	extent extentRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="extentChoiceGroup"> <xs:choice> <xs:element ref="extent"/> <xs:element ref="extentRef"/> </xs:choice> </xs:group></pre>

3.8.2 element extent

diagram	
namespace	http://purl.org/dc/terms/
type	extentType
used by	group extentChoiceGroup
source	<pre><xs:element name="extent" type="extentType"/></pre>

3.8.3 element extentRef

diagram	
namespace	http://purl.org/dc/terms/
type	extentRefType
used by	group extentChoiceGroup
source	<pre><xs:element name="extentRef" type="extentRefType"/></pre>

3.9 Property: *hasFormat*

Refines

dc:relation

Definition

The described resource pre-existed the referenced resource, which is essentially the same intellectual content presented in another format.


Comment

Examples of a Contributor include a person, an organisation, or a service. Typically, the name of a Contributor should be used to indicate the entity.

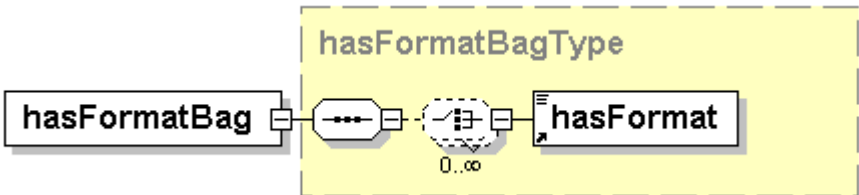
3.9.1 **group** *hasFormatChoiceGroup*

diagram	<p>The diagram shows a class named <code>hasFormatChoiceGroup</code> which is a choice group (indicated by a dashed line and a choice symbol). It contains four child elements: <code>hasFormat</code>, <code>hasFormatRef</code>, <code>hasFormatQVal</code>, and <code>hasFormatBag</code>. Each child element is represented by a box with a small icon indicating its cardinality and type.</p>
namespace	http://purl.org/dc/terms/
children	hasFormat hasFormatRef hasFormatQVal hasFormatBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="hasFormatChoiceGroup"> <xs:choice> <xs:element ref="hasFormat"/> <xs:element ref="hasFormatRef"/> <xs:element ref="hasFormatQVal"/> <xs:element ref="hasFormatBag"/> </xs:choice> </xs:group></pre>

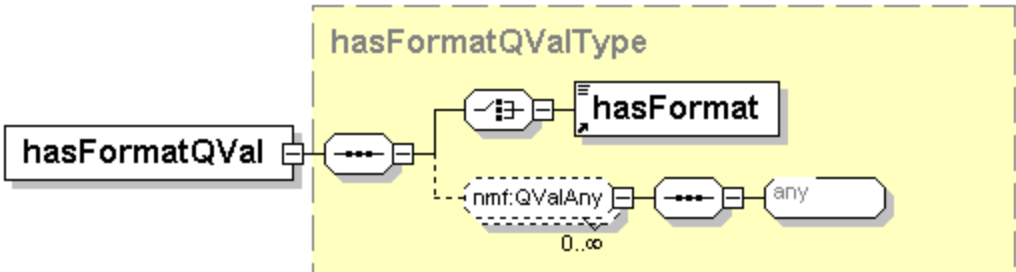
3.9.2 element hasFormat

diagram	
namespace	http://purl.org/dc/terms/
type	hasFormatType
used by	complexTypes group hasFormatBagType hasFormatQValType hasFormatChoiceGroup
source	<code><xs:element name="hasFormat" type="hasFormatType"/></code>

3.9.3 element hasFormatBag


diagram	
namespace	http://purl.org/dc/terms/
type	hasFormatBagType
children	hasFormat
used by	group hasFormatChoiceGroup
source	<code><xs:element name="hasFormatBag" type="hasFormatBagType"/></code>

3.9.4 element hasFormatQVal

diagram	
---------	--

namespace	http://purl.org/dc/terms/
type	hasFormatQValType
children	hasFormat
used by	group hasFormatChoiceGroup
source	<code><xs:element name="hasFormatQVal" type="hasFormatQValType"/></code>

3.9.5 element hasFormatRef

diagram	
namespace	http://purl.org/dc/terms/
type	hasFormatRefType
used by	group hasFormatChoiceGroup
source	<code><xs:element name="hasFormatRef" type="hasFormatRefType"/></code>

3.10 Property: hasPart

Refines

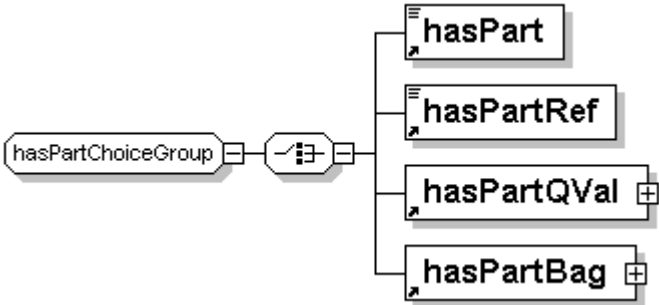
dc:relation

Definition

The described resource includes the referenced resource either physically or logically.


Comment

3.10.1 group hasPartChoiceGroup

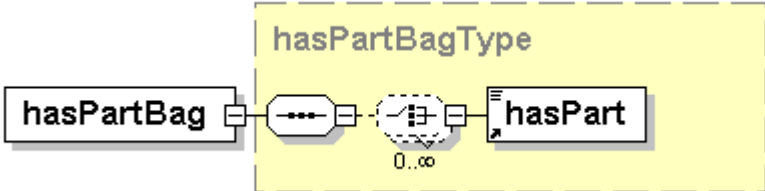
diagram	
namespace	http://purl.org/dc/terms/
children	hasPart hasPartRef hasPartQVal hasPartBag

used by	complexType BySchemaPropsType
source	<pre><xs:group name="hasPartChoiceGroup"> <xs:choice> <xs:element ref="hasPart"/> <xs:element ref="hasPartRef"/> <xs:element ref="hasPartQVal"/> <xs:element ref="hasPartBag"/> </xs:choice> </xs:group></pre>

3.10.2 element hasPart

diagram	
namespace	http://purl.org/dc/terms/
type	hasPartType
used by	complexTypes hasPartBagType hasPartQValType group hasPartChoiceGroup
source	<pre><xs:element name="hasPart" type="hasPartType"/></pre>

3.10.3 element hasPartBag

diagram	
namespace	http://purl.org/dc/terms/
type	hasPartBagType
children	hasPart
used by	group hasPartChoiceGroup
source	<pre><xs:element name="hasPartBag" type="hasPartBagType"/></pre>

3.10.4 element hasPartQVal

diagram	
namespace	http://purl.org/dc/terms/
type	hasPartQValType
children	hasPart
used by	group hasPartChoiceGroup
source	<code><xs:element name="hasPartQVal" type="hasPartQValType"/></code>

3.10.5 element hasPartRef

diagram	
namespace	http://purl.org/dc/terms/
type	hasPartRefType
used by	group hasPartChoiceGroup
source	<code><xs:element name="hasPartRef" type="hasPartRefType"/></code>

3.11 Property: *hasVersion*

Refines

dc:relation

Definition

The described resource has a version, edition, or adaptation, namely, the referenced resource.

Comment

3.11.1 **group** hasVersionChoiceGroup

diagram	<p>The diagram shows a box labeled 'hasVersionChoiceGroup' connected to a choice symbol (a circle with a vertical bar and a plus sign). This choice symbol is connected to four separate boxes: 'hasVersion', 'hasVersionRef', 'hasVersionQVal', and 'hasVersionBag'. Each of these four boxes has a plus sign in its top right corner, indicating that each element is optional within the choice.</p>
namespace	http://purl.org/dc/terms/
children	hasVersion hasVersionRef hasVersionQVal hasVersionBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="hasVersionChoiceGroup"> <xs:choice> <xs:element ref="hasVersion"/> <xs:element ref="hasVersionRef"/> <xs:element ref="hasVersionQVal"/> <xs:element ref="hasVersionBag"/> </xs:choice> </xs:group></pre>

3.11.2 **element** hasVersion

diagram	<p>The diagram shows a single box labeled 'hasVersion' with a plus sign in its top right corner, indicating it is an optional element.</p>
namespace	http://purl.org/dc/terms/
type	hasVersionType
used by	complexTypes hasVersionBagType hasVersionQValType group hasVersionChoiceGroup
source	<pre><xs:element name="hasVersion" type="hasVersionType"/></pre>

3.11.3 **element** hasVersionBag

diagram	<p>The diagram shows a box labeled 'hasVersionBag' connected to a bag symbol (a circle with three dots and a plus sign). This bag symbol is connected to a choice symbol (a circle with a vertical bar and a plus sign). The choice symbol is connected to a box labeled 'hasVersion'. The bag symbol has '0..∞' written below it, indicating that the bag can contain zero or an infinite number of instances of the 'hasVersion' element. The entire structure from the bag symbol to the 'hasVersion' box is enclosed in a dashed yellow box labeled 'hasVersionBagType'.</p>
namespace	http://purl.org/dc/terms/
type	hasVersionBagType
children	hasVersion

used by	group hasVersionChoiceGroup
source	<code><xs:element name="hasVersionBag" type="hasVersionBagType"/></code>

3.11.4 element `hasVersionQVal`

diagram	
namespace	<code>http://purl.org/dc/terms/</code>
type	hasVersionQValType
children	hasVersion
used by	group hasVersionChoiceGroup
source	<code><xs:element name="hasVersionQVal" type="hasVersionQValType"/></code>

3.11.5 element `hasVersionRef`

diagram	
namespace	<code>http://purl.org/dc/terms/</code>
type	hasVersionRefType
used by	group hasVersionChoiceGroup
source	<code><xs:element name="hasVersionRef" type="hasVersionRefType"/></code>

3.12 Property: *isformatOf*

Refines

dc:relation

Definition

The described resource is the same intellectual content of the referenced resource, but presented in another format.

Comment

3.12.1 **group** isFormatOfChoiceGroup

diagram	<p>The diagram shows a box labeled 'isFormatOfChoiceGroup' connected to a choice symbol (a circle with a vertical bar and a plus sign). This choice symbol is connected to four boxes representing the elements: 'isFormatOf', 'isFormatOfRef', 'isFormatOfQVal', and 'isFormatOfBag'.</p>
namespace	http://purl.org/dc/terms/
children	isFormatOf isFormatOfRef isFormatOfQVal isFormatOfBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="isFormatOfChoiceGroup"> <xs:choice> <xs:element ref="isFormatOf"/> <xs:element ref="isFormatOfRef"/> <xs:element ref="isFormatOfQVal"/> <xs:element ref="isFormatOfBag"/> </xs:choice> </xs:group></pre>

3.12.2 **element** isFormatOf

diagram	<p>The diagram shows a box labeled 'isFormatOf' with a small icon to its left.</p>
namespace	http://purl.org/dc/terms/
type	isFormatOfType
used by	complexTypes isFormatOfBagType isFormatOfQValType group isFormatOfChoiceGroup
source	<pre><xs:element name="isFormatOf" type="isFormatOfType"/></pre>

3.12.3 **element** isFormatOfBag

diagram	<p>The diagram shows a box labeled 'isFormatOfBag' connected to a bag symbol (a circle with three dots). This bag symbol is connected to a choice symbol (a circle with a vertical bar and a plus sign). The choice symbol is connected to a box labeled 'isFormatOf'. A yellow dashed box highlights the bag, choice, and isFormatOf elements, with the text 'isFormatOfBagType' above it and '0..∞' below the bag symbol.</p>
namespace	http://purl.org/dc/terms/
type	isFormatOfBagType
children	isFormatOf

used by	group isFormatOfChoiceGroup
source	<code><xs:element name="isFormatOfBag" type="isFormatOfBagType"/></code>

3.12.4 element isFormatOfQVal

diagram	
namespace	http://purl.org/dc/terms/
type	isFormatOfQValType
children	isFormatOf
used by	group isFormatOfChoiceGroup
source	<code><xs:element name="isFormatOfQVal" type="isFormatOfQValType"/></code>

3.12.5 element isFormatOfRef

diagram	
namespace	http://purl.org/dc/terms/
type	isFormatOfRefType
used by	group isFormatOfChoiceGroup
source	<code><xs:element name="isFormatOfRef" type="isFormatOfRefType"/></code>

3.13 Property: isPartOf

Refines

dc:relation

Definition

The described resource is a physical or logical part of the referenced resource.

Comment

3.13.1 **group** isPartOfChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	isPartOf isPartOfRef isPartOfQVal isPartOfBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="isPartOfChoiceGroup"> <xs:choice> <xs:element ref="isPartOf"/> <xs:element ref="isPartOfRef"/> <xs:element ref="isPartOfQVal"/> <xs:element ref="isPartOfBag"/> </xs:choice> </xs:group></pre>

3.13.2 **element** isPartOf

diagram	
namespace	http://purl.org/dc/terms/
type	isPartOfType
used by	complexTypes isPartOfBagType isPartOfQValType group isPartOfChoiceGroup
source	<pre><xs:element name="isPartOf" type="isPartOfType"/></pre>

3.13.3 **element** isPartOfBag

diagram	
namespace	http://purl.org/dc/terms/
type	isPartOfBagType
children	isPartOf

used by	group isPartOfChoiceGroup
source	<code><xs:element name="isPartOfBag" type="isPartOfBagType"/></code>

3.13.4 element isPartOfQVal

diagram	
namespace	http://purl.org/dc/terms/
type	isPartOfQValType
children	isPartOf
used by	group isPartOfChoiceGroup
source	<code><xs:element name="isPartOfQVal" type="isPartOfQValType"/></code>

3.13.5 element isPartOfRef

diagram	
namespace	http://purl.org/dc/terms/
type	isPartOfRefType
used by	group isPartOfChoiceGroup
source	<code><xs:element name="isPartOfRef" type="isPartOfRefType"/></code>

3.14 Property: isReferencedBy

Refines

dc:relation

Definition

The described resource is referenced, cited, or otherwise pointed to by the referenced resource.

Comment

3.14.1 **group** isReferencedByChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	isReferencedBy isReferencedByRef isReferencedByQVal isReferencedByBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="isReferencedByChoiceGroup"> <xs:choice> <xs:element ref="isReferencedBy"/> <xs:element ref="isReferencedByRef"/> <xs:element ref="isReferencedByQVal"/> <xs:element ref="isReferencedByBag"/> </xs:choice> </xs:group></pre>

3.14.2 **element** isReferencedBy

diagram	
namespace	http://purl.org/dc/terms/
type	isReferencedByType
used by	complexType isReferencedByBagType isReferencedByQValType group isReferencedByChoiceGroup
source	<pre><xs:element name="isReferencedBy" type="isReferencedByType"/></pre>

3.14.3 **element** isReferencedByBag

diagram	
namespace	http://purl.org/dc/terms/
type	isReferencedByBagType
children	isReferencedBy

used by	group isReferencedByChoiceGroup
source	<code><xs:element name="isReferencedByBag" type="isReferencedByBagType"/></code>

3.14.4 element isReferencedByQVal

diagram	
namespace	http://purl.org/dc/terms/
type	isReferencedByQValType
children	isReferencedBy
used by	group isReferencedByChoiceGroup
source	<code><xs:element name="isReferencedByQVal" type="isReferencedByQValType"/></code>

3.14.5 element isReferencedByRef

diagram	
namespace	http://purl.org/dc/terms/
type	isReferencedByRefType
used by	group isReferencedByChoiceGroup
source	<code><xs:element name="isReferencedByRef" type="isReferencedByRefType"/></code>

3.15 isReplacedBy

Refines

dc:relation

Definition

the described resource is supplanted, displaced, or superceded by the referenced resource.

Comment

3.15.1 **group** isReplacedByChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	isReplacedBy isReplacedByRef isReplacedByQVal isReplacedByBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="isReplacedByChoiceGroup"> <xs:choice> <xs:element ref="isReplacedBy"/> <xs:element ref="isReplacedByRef"/> <xs:element ref="isReplacedByQVal"/> <xs:element ref="isReplacedByBag"/> </xs:choice> </xs:group></pre>

3.15.2 **element** isReplacedBy

diagram	
namespace	http://purl.org/dc/terms/
type	isReplacedByType
used by	complexType isReplacedByBagType isReplacedByQValType group isReplacedByChoiceGroup
source	<pre><xs:element name="isReplacedBy" type="isReplacedByType"/></pre>

3.15.3 element isReplacedByBag

diagram	
namespace	http://purl.org/dc/terms/
type	isReplacedByBagType
children	isReplacedBy
used by	group isReplacedByChoiceGroup
source	<code><xs:element name="isReplacedByBag" type="isReplacedByBagType"/></code>

3.15.4 element isReplacedByQVal

diagram	
namespace	http://purl.org/dc/terms/
type	isReplacedByQValType
children	isReplacedBy
used by	group isReplacedByChoiceGroup
source	<code><xs:element name="isReplacedByQVal" type="isReplacedByQValType"/></code>

3.15.5 element isReplacedByRef

diagram	
namespace	http://purl.org/dc/terms/
type	isReplacedByRefType
used by	group isReplacedByChoiceGroup

source	<code><xs:element name="isReplacedByRef" type="isReplacedByRefType"/></code>
--------	--

3.16 Property: *isRequiredBy*

Refines

dc:relation

Definition

The described resource is required by the referenced resource, either physically or logically.

Comment

3.16.1 group *isRequiredByChoiceGroup*

diagram	<pre> classDiagram class isRequiredByChoiceGroup class isRequiredBy class isRequiredByRef class isRequiredByQVal class isRequiredByBag isRequiredByChoiceGroup -- isRequiredBy isRequiredByChoiceGroup -- isRequiredByRef isRequiredByChoiceGroup -- isRequiredByQVal isRequiredByChoiceGroup -- isRequiredByBag </pre>
namespace	http://purl.org/dc/terms/
children	isRequiredBy isRequiredByRef isRequiredByQVal isRequiredByBag
used by	complexType BySchemaPropsType
source	<pre> <xs:group name="isRequiredByChoiceGroup"> <xs:choice> <xs:element ref="isRequiredBy"/> <xs:element ref="isRequiredByRef"/> <xs:element ref="isRequiredByQVal"/> <xs:element ref="isRequiredByBag"/> </xs:choice> </xs:group> </pre>

3.16.2 element *isRequiredBy*

diagram	<pre> classDiagram class isRequiredBy </pre>
namespace	http://purl.org/dc/terms/
type	isRequiredByType
used by	complexType isRequiredByBagType isRequiredByQValType group isRequiredByChoiceGroup

source	<code><xs:element name="isRequiredBy" type="isRequiredByType"/></code>
--------	--

3.16.3 element isRequiredByBag

diagram	
namespace	http://purl.org/dc/terms/
type	isRequiredByBagType
children	isRequiredBy
used by	group isRequiredByChoiceGroup
source	<code><xs:element name="isRequiredByBag" type="isRequiredByBagType"/></code>

3.16.4 element isRequiredByQVal

diagram	
namespace	http://purl.org/dc/terms/
type	isRequiredByQValType
children	isRequiredBy
used by	group isRequiredByChoiceGroup
source	<code><xs:element name="isRequiredByQVal" type="isRequiredByQValType"/></code>

3.16.5 element isRequiredByRef

diagram	
namespace	http://purl.org/dc/terms/
type	isRequiredByRefType

used by	group isRequiredByChoiceGroup
source	<code><xs:element name="isRequiredByRef" type="isRequiredByRefType"/></code>

3.17 Property: *issued*

Refines

dc:date

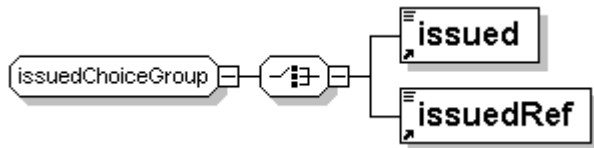
Definition

Date of formal issuance (e.g., publication) of the resource.

Comment

Examples of a Contributor include a person, an organisation, or a service. Typically, the name of a Contributor should be used to indicate the entity.


3.17.1 group issuedChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	issued issuedRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="issuedChoiceGroup"> <xs:choice> <xs:element ref="issued"/> <xs:element ref="issuedRef"/> </xs:choice> </xs:group></pre>

3.17.2 element issued

diagram	
namespace	http://purl.org/dc/terms/
type	issuedType
used by	group issuedChoiceGroup
source	<code><xs:element name="issued" type="issuedType"/></code>

3.17.3 element issuedRef

diagram	
namespace	http://purl.org/dc/terms/
type	issuedRefType
used by	group issuedChoiceGroup
source	<code><xs:element name="issuedRef" type="issuedRefType"/></code>

3.18 Property: *isVersionOf*

Refines

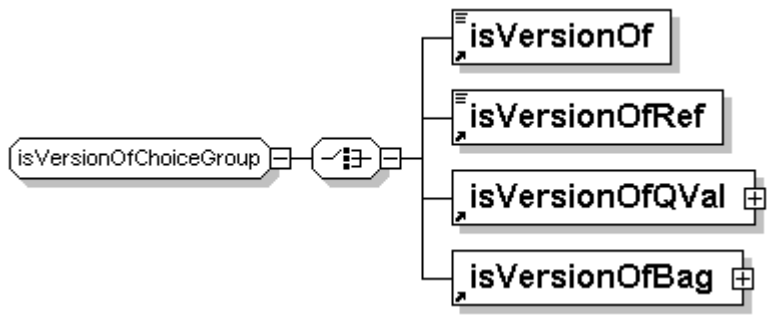
dc:relation

Definition

The described resource is a version, edition, or adaptation of the referenced resource. Changes in version imply substantive changes in content rather than differences in format.

Comment

3.18.1 group isVersionOfChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	isVersionOf isVersionOfRef isVersionOfQVal isVersionOfBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="isVersionOfChoiceGroup"> <xs:choice> <xs:element ref="isVersionOf"/> <xs:element ref="isVersionOfRef"/> <xs:element ref="isVersionOfQVal"/> <xs:element ref="isVersionOfBag"/> </xs:choice> </xs:group></pre>

	<pre></xs:choice> </xs:group></pre>
--	---

3.18.2 element isVersionOf

diagram	
namespace	http://purl.org/dc/terms/
type	isVersionOfType
used by	complexTypes isVersionOfBagType isVersionOfQValType group isVersionOfChoiceGroup
source	<code><xs:element name="isVersionOf" type="isVersionOfType"/></code>

3.18.3 element isVersionOfBag


diagram	
namespace	http://purl.org/dc/terms/
type	isVersionOfBagType
children	isVersionOf
used by	group isVersionOfChoiceGroup
source	<code><xs:element name="isVersionOfBag" type="isVersionOfBagType"/></code>

3.18.4 element isVersionOfQVal

diagram	
namespace	http://purl.org/dc/terms/
type	isVersionOfQValType
children	isVersionOf
used by	group isVersionOfChoiceGroup

source	<code><xs:element name="isVersionOfQVal" type="isVersionOfQValType"/></code>
--------	--

3.18.5 element isVersionOfRef

diagram	
namespace	http://purl.org/dc/terms/
type	isVersionOfRefType
used by	group isVersionOfChoiceGroup
source	<code><xs:element name="isVersionOfRef" type="isVersionOfRefType"/></code>

3.19 Property: medium

Refines

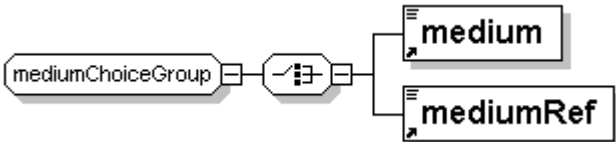
dc:format

Definition

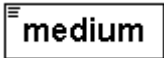
The material or physical carrier of the resource.

Comment

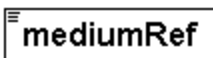
3.19.1 group mediumChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	medium mediumRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="mediumChoiceGroup"> <xs:choice> <xs:element ref="medium"/> <xs:element ref="mediumRef"/> </xs:choice> </xs:group></pre>

3.19.2 element medium

diagram	
namespace	http://purl.org/dc/terms/
type	mediumType
used by	group mediumChoiceGroup
source	<code><xs:element name="medium" type="mediumType"/></code>

3.19.3 element mediumRef

diagram	
namespace	http://purl.org/dc/terms/
type	mediumRefType
used by	group mediumChoiceGroup
source	<code><xs:element name="mediumRef" type="mediumRefType"/></code>

3.20 *Property: modified*

Refines

dc:date

Definition

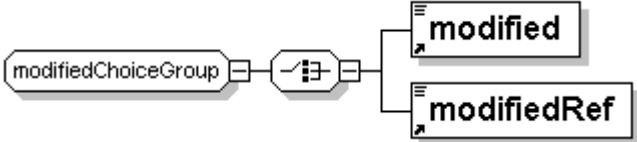
Date on which the resource was changed.

Comment

Practice


This property is specified using the xs:datetime syntax which conforms to ISO 8601.

3.20.1 group modifiedChoiceGroup


diagram	
namespace	http://purl.org/dc/terms/
children	modified modifiedRef

used by	complexType BySchemaPropsType
source	<pre><xs:group name="modifiedChoiceGroup"> <xs:choice> <xs:element ref="modified"/> <xs:element ref="modifiedRef"/> </xs:choice> </xs:group></pre>

3.20.2 element modified

diagram	
namespace	http://purl.org/dc/terms/
type	modifiedType
used by	group modifiedChoiceGroup
source	<pre><xs:element name="modified" type="modifiedType"/></pre>

3.20.3 element modifiedRef

diagram	
namespace	http://purl.org/dc/terms/
type	modifiedRefType
used by	group modifiedChoiceGroup
source	<pre><xs:element name="modifiedRef" type="modifiedRefType"/></pre>

3.21 *Property: references*

Refines

dc:relation

Definition

The described resource references, cites, or otherwise points to the referenced resource..

Comment

3.21.1 **group** referencesChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	references referencesRef referencesQVal referencesBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="referencesChoiceGroup"> <xs:choice> <xs:element ref="references"/> <xs:element ref="referencesRef"/> <xs:element ref="referencesQVal"/> <xs:element ref="referencesBag"/> </xs:choice> </xs:group></pre>

3.21.2 **element** references

diagram	
namespace	http://purl.org/dc/terms/
type	referencesType
used by	complexType referencesBagType referencesQValType group referencesChoiceGroup
source	<pre><xs:element name="references" type="referencesType"/></pre>

3.21.3 **element** referencesBag

diagram	
namespace	http://purl.org/dc/terms/
type	referencesBagType
children	references

used by	group referencesChoiceGroup
source	<code><xs:element name="referencesBag" type="referencesBagType"/></code>

3.21.4 element referencesQVal

diagram	
namespace	http://purl.org/dc/terms/
type	referencesQValType
children	references
used by	group referencesChoiceGroup
source	<code><xs:element name="referencesQVal" type="referencesQValType"/></code>

3.21.5 element referencesRef

diagram	
namespace	http://purl.org/dc/terms/
type	referencesRefType
used by	group referencesChoiceGroup
source	<code><xs:element name="referencesRef" type="referencesRefType"/></code>

3.22 Property: replaces

Refines

dc:relation

Definition

The described resource supplants, displaces, or supersedes the referenced resource.

Comment

3.22.1 **group** replacesChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	replaces replacesRef replacesQVal replacesBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="replacesChoiceGroup"> <xs:choice> <xs:element ref="replaces"/> <xs:element ref="replacesRef"/> <xs:element ref="replacesQVal"/> <xs:element ref="replacesBag"/> </xs:choice> </xs:group></pre>

3.22.2 **element** replaces

diagram	
namespace	http://purl.org/dc/terms/
type	replacesType
used by	complexTypes replacesBagType replacesQValType group replacesChoiceGroup
source	<pre><xs:element name="replaces" type="replacesType"/></pre>

3.22.3 **element** replacesBag

diagram	
namespace	http://purl.org/dc/terms/
type	replacesBagType
children	replaces

used by	group replacesChoiceGroup
source	<code><xs:element name="replacesBag" type="replacesBagType"/></code>

3.22.4 element replacesQVal

diagram	<p>The diagram shows the structure of the <code>replacesQVal</code> element. It is a sequence of elements: first, a <code>replaces</code> element; then, a choice between <code>nmf:QValAny</code> (with cardinality <code>0..∞</code>) and <code>any</code>.</p>
namespace	http://purl.org/dc/terms/
type	replacesQValType
children	replaces
used by	group replacesChoiceGroup
source	<code><xs:element name="replacesQVal" type="replacesQValType"/></code>

3.22.5 element replacesRef

diagram	<p>The diagram shows the structure of the <code>replacesRef</code> element, which is a single <code>replacesRef</code> element.</p>
namespace	http://purl.org/dc/terms/
type	replacesRefType
used by	group replacesChoiceGroup
source	<code><xs:element name="replacesRef" type="replacesRefType"/></code>

3.23 Property: requires

Refines

dc:relation

Definition

The described resource requires the referenced resource to support its function, delivery, or coherence of content.

Comment

3.23.1 **group** requiresChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	requires requiresRef requiresQVal requiresBag
used by	complexType BySchemaPropsType
source	<pre><xs:group name="requiresChoiceGroup"> <xs:choice> <xs:element ref="requires"/> <xs:element ref="requiresRef"/> <xs:element ref="requiresQVal"/> <xs:element ref="requiresBag"/> </xs:choice> </xs:group></pre>

3.23.2 **element** requires

diagram	
namespace	http://purl.org/dc/terms/
type	requiresType
used by	complexType requiresBagType requiresQValType group requiresChoiceGroup
source	<pre><xs:element name="requires" type="requiresType"/></pre>

3.23.3 **element** requiresBag

diagram	
namespace	http://purl.org/dc/terms/
type	requiresBagType
children	requires

used by	group requiresChoiceGroup
source	<code><xs:element name="requiresBag" type="requiresBagType"/></code>

3.23.4 element `requiresQVal`

diagram	
namespace	http://purl.org/dc/terms/
type	requiresQValType
children	requires
used by	group requiresChoiceGroup
source	<code><xs:element name="requiresQVal" type="requiresQValType"/></code>

3.23.5 element `requiresRef`

diagram	
namespace	http://purl.org/dc/terms/
type	requiresRefType
used by	group requiresChoiceGroup
source	<code><xs:element name="requiresRef" type="requiresRefType"/></code>

3.24 Property: *spatial*

Refines

dc:coverage

Definition

Spatial characteristics of the intellectual content of the resource.

Comment

3.24.1 group spatialChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	spatial spatialRef spatialQVal
used by	complexType BySchemaPropsType
source	<pre><xs:group name="spatialChoiceGroup"> <xs:choice> <xs:element ref="spatial"/> <xs:element ref="spatialRef"/> <xs:element ref="spatialQVal"/> </xs:choice> </xs:group></pre>

3.24.2 element spatial

diagram	
namespace	http://purl.org/dc/terms/
type	spatialType
used by	complexType spatialQValType group spatialChoiceGroup
source	<pre><xs:element name="spatial" type="spatialType"/></pre>

3.24.3 element spatialQVal

diagram	
namespace	http://purl.org/dc/terms/
type	spatialQValType
children	spatial
used by	group spatialChoiceGroup

source	<code><xs:element name="spatialQVal" type="spatialQValType"/></code>
--------	--

3.24.4 element spatialRef

diagram	
namespace	http://purl.org/dc/terms/
type	spatialRefType
used by	group spatialChoiceGroup
source	<code><xs:element name="spatialRef" type="spatialRefType"/></code>

3.25 Property: tableOfContents

Refines

description

Definition

A list of subunits of the content of the resource.

Comment

3.25.1 group tableOfContentsChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	tableOfContents tableOfContentsRef tableOfContentsQVal tableOfContentsAlt tableOfContentsAnyXML
used by	complexType BySchemaPropsType
source	<code><xs:group name="tableOfContentsChoiceGroup"> <xs:choice></code>

	<pre> <xs:element ref="tableOfContents"/> <xs:element ref="tableOfContentsRef"/> <xs:element ref="tableOfContentsQVal"/> <xs:element ref="tableOfContentsAlt"/> <xs:element ref="tableOfContentsAnyXML"/> </xs:choice> </xs:group> </pre>
--	---

3.25.2 element tableOfContents

diagram	
namespace	http://purl.org/dc/terms/
type	tableOfContentsType
used by	complexTypes tableOfContentsAltType tableOfContentsQValType group tableOfContentsChoiceGroup
source	<xs:element name="tableOfContents" type="tableOfContentsType"/>

3.25.3 element tableOfContentsAlt

diagram	
namespace	http://purl.org/dc/terms/
type	tableOfContentsAltType
children	tableOfContents
used by	group tableOfContentsChoiceGroup
source	<xs:element name="tableOfContentsAlt" type="tableOfContentsAltType"/>

3.25.4 element tableOfContentsAnyXML

diagram	
namespace	http://purl.org/dc/terms/
type	tableOfContentsAnyXMLType
used by	group tableOfContentsChoiceGroup

source	<code><xs:element name="tableOfContentsAnyXML" type="tableOfContentsAnyXMLType"/></code>
--------	--

3.25.5 element tableOfContentsQVal

diagram	
namespace	http://purl.org/dc/terms/
type	tableOfContentsQValType
children	tableOfContents
used by	group tableOfContentsChoiceGroup
source	<code><xs:element name="tableOfContentsQVal" type="tableOfContentsQValType"/></code>

3.25.6 element tableOfContentsRef

diagram	
namespace	http://purl.org/dc/terms/
type	tableOfContentsRefType
used by	group tableOfContentsChoiceGroup
source	<code><xs:element name="tableOfContentsRef" type="tableOfContentsRefType"/></code>

3.26 Property: temporal

Refines

dc:coverage

Definition

Temporal characteristics of the intellectual content of the resource.

Comment

3.26.1 group temporalChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	temporal temporalRef temporalQVal
used by	complexType BySchemaPropsType
source	<pre><xs:group name="temporalChoiceGroup"> <xs:choice> <xs:element ref="temporal"/> <xs:element ref="temporalRef"/> <xs:element ref="temporalQVal"/> </xs:choice> </xs:group></pre>

3.26.2 element temporal


diagram	
namespace	http://purl.org/dc/terms/
type	temporalType
used by	complexType temporalQValType group temporalChoiceGroup
source	<pre><xs:element name="temporal" type="temporalType"/></pre>

3.26.3 element temporalQVal

diagram	
namespace	http://purl.org/dc/terms/
type	temporalQValType
children	temporal
used by	group temporalChoiceGroup

source	<code><xs:element name="temporalQVal" type="temporalQValType"/></code>
--------	--

3.26.4 element temporalRef

diagram	
namespace	http://purl.org/dc/terms/
type	temporalRefType
used by	group temporalChoiceGroup
source	<code><xs:element name="temporalRef" type="temporalRefType"/></code>

3.27 Property: *valid*

Refines

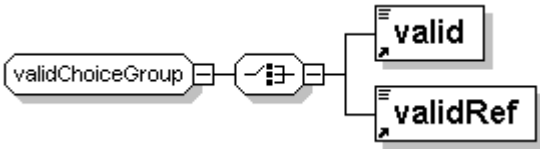
dc:date

Definition


Date (often a range) of validity of a resource.

Comment


3.27.1 group validChoiceGroup

diagram	
namespace	http://purl.org/dc/terms/
children	valid validRef
used by	complexType BySchemaPropsType
source	<pre><xs:group name="validChoiceGroup"> <xs:choice> <xs:element ref="valid"/> <xs:element ref="validRef"/> </xs:choice> </xs:group></pre>

3.27.2 element valid

diagram	
namespace	http://purl.org/dc/terms/
type	validType
used by	group validChoiceGroup
source	<code><xs:element name="valid" type="validType"/></code>

3.27.3 element validRef

diagram	
namespace	http://purl.org/dc/terms/
type	validRefType
used by	group validChoiceGroup
source	<code><xs:element name="validRef" type="validRefType"/></code>

Appendix 1. References

[DC-HTML]

“Encoding Dublin Core Metadata in HTML” J. Kunze,, Dec 1999
Available at <http://www.ietf.org/rfc/rfc2731.txt/>

[DC-RDF]

“Guidance on expressing the Dublin Core within the Resource Description Framework (RDF)”, Eric Miller, Paul Miller and Dan Brickley. Dublin Core Metadata Initiative Draft, July 1999.
Available at <http://www.ukoln.ac.uk/metadata/resources/dc/datamodel/WD-dc-rdf/>

[DC-RDF-Simple]

“Expressing Simple Dublin Core in RDF/XML”, Dave Beckett, Eric Miller and Dan Brickley, Dublin Core Metadata Initiative Proposed Recommendation, November 28th, 2001.
Available at <http://dublincore.org/documents/dcmes-xml/>

[DC-USAGE]

“Using Dublin Core”, Dianne Hilman, Dublin Core Metadata Initiative Recommendation, April 4th, 2001
<http://dublincore.org/documents/2001/04/12/usageguide/>

[DC-XML]

“Guidelines for implementing Dublin Core in XML” Andy Powell,
Available at <http://www.ukoln.ac.uk/metadata/dcmi/dc-xml-guidelines/>

[DCES]

The specification for the Dublin Core Element Set can be found at: <http://dublincore.org/documents/dces/>.

[DCMI]

The Dublin Core Metadata Initiative can be found at: <http://dublincore.org/>.

[DCMI-NS]

“Namespace Policy for the Dublin Core Metadata Initiative”, Andy Powell and Eric Wagner, Dublin Core Metadata Initiative Recommendation, October 26th, 2001.
Available at <http://dublincore.org/documents/dcmi-namespace/>

[DCQ]

“Dublin Core Qualifiers”, Dublin Core Metadata Initiative Recommendation, July 11th, 200.
Available at <http://dublincore.org/documents/dcmes-qualifiers/>

[DCQ-RDF]

“Expressing Qualified Dublin Core in RDF / XML”, Stefan Kokkelink and Roland Schwäenzl, Dublin Core Metadata Initiative Proposed Recommendation, August 29th, 2001.
Available at <http://dublincore.org/documents/dcq-rdf-xml/>

[MIME]

Internet Media Types.

Available at: <http://www.isi.edu/in-notes/iana/assignments/media-types/media-types>

[MPV-Core]

"MPV Core Specification 1.0"; OSTA, 2002.

Available at <http://www.osta.org/mpv/>

[NMF]

"Normalized Metadata Format Specification 1.0"; OSTA, 2002.

Available at <http://www.osta.org/mpv/>

[OSTA-WEB]

"OSTA MulitPhoto/Video Initiative", 2002,

Available at <http://www.osta.org/mpv/>

[PRISM]

"PRISM: Publishing Requirements for Industry Standard Metadata", Prism working group, April 9th, 2001.

Available at <http://www.prismstandard.org/techdev/prismspec1.asp>

[RFC2119]

"Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, IETF RFC 2119

<http://www.ietf.org/rfc/rfc2119.txt>

[RDF]

"Resource Description Framework (RDF) Model and Syntax Specification", Ora Lassila and Ralph R. Swick. W3C Recommendation 22 February 1999,

Available at <http://www.w3.org/TR/REC-rdf-syntax/>

[RDFschema]

"Resource Description Framework (RDF) Schema Specification", Dan Brickley and R.V. Guha. W3C Proposed Recommendation 03 March 1999,

Available at <http://www.w3.org/TR/PR-rdf-schema/>

[RSS1]

"Rich Site Summary 1.0", RSS Working Group, December 6th, 2000.

Available at <http://purl.org/rss/1.0/spec>.

[URI]

"Uniform Resource Identifiers (URI): Generic Syntax", T. Berners-Lee, R. Fielding, L. Masinter, August 1998. Note that RFC 2396 updates [RFC1738] and [RFC1808].

[UCS-2]

16-bit encoding of ISO 10646, commonly known as the Unicode character set.

[UTF-8]

Yergeau, F., "UTF-8, a transformation format of ISO 10646", RFC 2279, January 1998.

[W3C-NSURI]

"URIs for W3C namespaces". Policy and administrative issue for W3C, Oct. 1999.

Available at <http://www.w3.org/1999/10/nsuri>

[XML10]

"Extensible Markup Language (XML) 1.0" T. Bray, J. Paoli and C.M. Sperberg-McQueen. W3C Recommendation 10 February 1998 ,

Available at <http://www.w3.org/TR/REC-xml>

[XML-NS]

"Namespaces in XML", Tim Bray, Dave Hollander, Andrew Layman. W3C Recommendation 14 January 1999,

Available at <http://www.w3.org/TR/REC-xml-names>

[XMP-FW]

"XMP – Extensible Metadata Platform 14 Sept 01", Copyright 2001 Adobe Inc,
Available at <http://partners.adobe.com/asn/developer/xmp/download/docs/MetadataFramework.pdf>

[XSHEMA]

"XML Schema, XML Schema Part 1: Structures". W3C Recommendation 2 May, 2001.
Available at <http://www.w3.org/TR/xmlschema-1/>

[XSHEMA2]

"XML Schema, XML Schema Part 2: Datatypes". W3C Recommendation, 2 May, 2001.
Available at <http://www.w3.org/TR/xmlschema-2/>

[XSL]

"XSL Transformations (XSLT) Version 1.0", W3C Recommendation, 16 November, 1999.
Available at <http://www.w3.org/TR/xsl/>

[XMLBASE]

"XML Base", Jonathan Marsh. W3C Recommendation, June 27th, 2001.
Available at <http://www.w3.org/TR/xmlbase/>