IPTC Standards

EventsML G2

Specification Version 1.0
Power Conformance Level

Document Revision 2

Copyrights

Acknowledgements

About the Standard

Specification Versioning History

Version	Issue Date	Approved by	Remarks
1.0	2008-01-31	IPTC Standards Committee	

Document Revision History

Revision	Issue Date	Author (revised by)	Remarks
1	2008-01-31	Michael Steidl	
2	2008-02-25	Michael Steidl	Errata fixed.

About this Document

This document specifies the IPTC news exchange standard EventsML-G2 which is a conceptual and processing model making freely available the IPTC knowledge of the most effective ways to structure, describe, manage and exchange newsworthy information about events.

Status of this Document

This document is under the governance of the IPTC EventsML-G2 Working Group (EventsML-G2 WG) of the IPTC News Content Working Party (NCT WP).

This is a specification document which was endorsed by the IPTC members and may be updated, replaced or obsoleted by other documents at any time.

Public versions of this document and of related IPTC documents are available at:

http://www.iptc.org/std/EventsML-G2/1.0/

Public comments should be sent to the forum and mailing list at:

http://tech.groups.yahoo.com/group/eventsml-g2

A page with all errata not covered by the latest version of the EventsML-G2 specification is available at:

http://www.iptc.org/goto?EventsML-G2-1.0Errata

The Full Set of Specification Documents

The full set of specification documents for EventsML-G2 1.0 consists of (file names are added, # is to be substituted by the most current document revision number):

♦ This Specification document - EventsML-G2_1.0_Specification_#.pdf

♦ These XML Schema files:

For the Core Conformance Level (see **Conformance Levels** on page 13):

● EventsML-G2_1.0-spec-Framework-Core.xsd

● EventsML-G2_1.0-spec-Framework-Core_#.xsd

● EventsML-G2_1.0-spec-ConceptItem-Core.xsd

● EventsML-G2_1.0-spec-ConceptItem-Core_#.xsd

● EventsML-G2_1.0-spec-KnowledgeItem-Core.xsd

● EventsML-G2_1.0-spec-KnowledgeItem-Core_#.xsd

● NAR_1.1-spec-Framework-Core.xsd

● NAR_1.1-spec-Framework-Core_1.xsd

For the Power Conformance Level (see **Conformance Levels** on page 13):

● EventsML-G2_1.0-spec-Framework-Power.xsd

● EventsML-G2_1.0-spec-Framework-Power_#.xsd

● EventsML-G2_1.0-spec-ConceptItem-Power.xsd

● EventsML-G2_1.0-spec-ConceptItem-Power_#.xsd

● EventsML-G2_1.0-spec-KnowledgeItem-Power.xsd

● EventsML-G2_1.0-spec-KnowledgeItem-Power_#.xsd

● NAR_1.1-spec-Framework-Power.xsd

● NAR_1.1-spec-Framework-Power_1.xsd

All files above can be obtained from:

http://www.iptc.org/std/EventsML-G2/1.0/specification/

XML Schemas for the generic G2 Package Item and the News Message can be obtained from:

http://www.iptc.org/std/NAR/1.1/specification/

Note on the XML Schema File Names

XML Schemas are revised for two reasons:

♦ The EventsML-G2 specifications have been changed: this results in an new version of the standards, this will be reflected by a new path to files and a new standard version number like EventsML-G2_1.9

♦ The XML Schema has been edited a) to fix errors and b) to change non-normative parts, like the wording of an element's annotation: this is reflected by a new revision number at the end of the file name like the "8" in EventsML-G2_1.0-spec-Framework-Core_8.xsd.

The XML Schema files without the document revision number (e.g. "_8") at the end of the file name are true copies of the latest document revision. This allows to apply a persistent reference to the latest XML Schema file version regardless of any edits of the document.

License Agreement

This document is issued under the *Non-Exclusive License Agreement for International Press Telecommunications Council Specifications and Related Documentation*

Important: International Press Telecommunications Council (IPTC) standard specifications for news (the Specifications) and supporting software, documentation, technical reports, web sites and other material related to the Specifications (the Materials) including the document accompanying this license (the Document), whether in a paper or electronic format, are made available to you subject to the terms stated below. By obtaining, using and/or copying the Specifications or Materials, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

1. The Specifications and Materials are licensed for use only on the condition that you agree to be bound by the terms of this license. Subject to this and other licensing requirements contained herein, you may, on a non-exclusive basis, use the Specifications and Materials.

2. The IPTC openly provides the Specifications and Materials for voluntary use by individuals, partnerships, companies, corporations, organizations and any other entity for use at the entity's own risk. This disclaimer, license and release is intended to apply to the IPTC, its officers, directors, agents, representatives, members, contributors, affiliates, contractors, or co-venturers acting jointly or severally.

3. The Document and translations thereof may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the copyright and license notices and references to the IPTC appearing in the Document and the terms of this Specifications License Agreement are included on all such copies and derivative works. Further, upon the receipt of written permission from the IPTC, the Document may be modified for the purpose of developing applications that use IPTC Specifications or as required to translate the Document into languages other than English.

4. Any use, duplication, distribution, or exploitation of the Document and Specifications and Materials in any manner is at your own risk.

5. NO WARRANTY, EXPRESSED OR IMPLIED, IS MADE REGARDING THE ACCURACY, ADEQUACY, COMPLETENESS, LEGALITY, RELIABILITY OR USEFULNESS OF ANY INFORMATION CONTAINED IN THE DOCUMENT OR IN ANY SPECIFICATION OR OTHER PRODUCT OR SERVICE PRODUCED OR SPONSORED BY THE IPTC. THE DOCUMENT AND THE INFORMATION CONTAINED HEREIN AND INCLUDED IN ANY SPECIFICATION OR OTHER PRODUCT OR SERVICE OF THE IPTC IS PROVIDED ON AN "AS IS" BASIS. THE IPTC DISCLAIMS ALL WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY ACTUAL OR ASSERTED WARRANTY OF NON-INFRINGEMENT OF PROPRIETARY RIGHTS, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. NEITHER THE IPTC NOR ITS CONTRIBUTORS SHALL BE HELD LIABLE FOR ANY IMPROPER OR INCORRECT USE OF INFORMATION. NEITHER THE IPTC NOR ITS CONTRIBUTORS ASSUME ANY RESPONSIBILITY FOR ANYONE'S USE OF INFORMATION PROVIDED BY THE IPTC. IN NO EVENT SHALL THE IPTC OR ITS CONTRIBUTORS BE LIABLE TO ANYONE FOR DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO, COMPENSATORY DAMAGES, LOST PROFITS, LOST DATA OR ANY FORM OF SPECIAL, INCIDENTAL, INDIRECT, CONSEQUENTIAL OR PUNITIVE DAMAGES OF ANY KIND WHETHER BASED ON BREACH OF CONTRACT OR WARRANTY, TORT, PRODUCT LIABILITY OR OTHERWISE.

6. The IPTC takes no position regarding the validity or scope of any Intellectual Property or other rights that might be claimed to pertain to the implementation or use of the technology described in the Document or the extent to which any license under such rights might or might not be available. The IPTC does not represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication, assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of the Specifications and Materials, can be obtained from the Managing Director of the IPTC.

7. By using the Specifications and Materials including the Document in any manner or for any purpose, you release the IPTC from all liabilities, claims, causes of action, allegations, losses, injuries, damages, or detriments of any nature arising from or relating to the use of the Specifications, Materials or any portion thereof. You further agree not to file a lawsuit, make a claim, or take any other formal or informal legal action against the IPTC, resulting from your acquisition, use, duplication, distribution, or exploitation of the Specifications, Materials or any portion thereof. Finally, you hereby agree that the IPTC is not liable for

any direct, indirect, special or consequential damages arising from or relating to your acquisition, use, duplication, distribution, or exploitation of the Specifications, Materials or any portion thereof.

8. Specifications and Materials may be downloaded or copied provided that ALL copies retain the ownership, copyright and license notices.

9. Materials may not be edited, modified, or presented in a context that creates a misleading or false impression or statement as to the positions, actions, or statements of the IPTC.

10. The name and trademarks of the IPTC may not be used in advertising, publicity, or in relation to products or services and their names without the specific, written prior permission of the IPTC. Any permitted use of the trademarks of the IPTC, whether registered or not, shall be accompanied by an appropriate mark and attribution, as agreed with the IPTC.

11. Specifications may be extended by both members and non-members to provide additional functionality (Extended Specifications) provided that there is a clear recognition of the IPTC IP and its ownership in the Extended Specifications and the related documentation and provided that the extensions are clearly identified and provided that a perpetual license is granted by the creator of the Extended Specifications for other members and non-members to use the Extended Specifications and to continue extensions of the Extended Specifications. The IPTC does not waive any of its rights in the Specifications and Materials in this context. The Extended Specifications may be considered the intellectual property of their creator. The IPTC expressly disclaims any responsibility for damage caused by an extension to the Specifications.

12. Specifications and Materials may be included in derivative work of both members and non-members provided that there is a clear recognition of the IPTC IP and its ownership in the derivative work and its related documentation. The IPTC does not waive any of its rights in the Specifications and Materials in this context. Derivative work in its entirety may be considered the intellectual property of the creator of the work .The IPTC expressly disclaims any responsibility for damage caused when its IP is used in a derivative context.

13. This Specifications License Agreement is perpetual subject to your conformance to the terms of this Agreement. The IPTC may terminate this Specifications License Agreement immediately upon your breach of this Agreement and, upon such termination you will cease all use, duplication, distribution, and/or exploitation in any manner of the Specifications and Materials.

14. This Specifications License Agreement reflects the entire agreement of the parties regarding the subject matter hereof and supersedes all prior agreements or representations regarding such matters, whether written or oral. To the extent any portion or provision of this Specifications License Agreement is found to be illegal or unenforceable, then the remaining provisions of this Specifications License Agreement will remain in full force and effect and the illegal or unenforceable provision will be construed to give it such effect as it may properly have that is consistent with the intentions of the parties.

15. This Specifications License Agreement may only be modified in writing signed by an authorized representative of the IPTC.

16. This Specifications License Agreement is governed by the law of United Kingdom, as such law is applied to contracts made and fully performed in the United Kingdom. Any disputes arising from or relating to this Specifications License Agreement will be resolved in the courts of the United Kingdom. You consent to the jurisdiction of such courts over you and covenant not to assert before such courts any objection to proceeding in such forums.

IF YOU DO NOT AGREE TO THESE TERMS YOU MUST CEASE ALL USE OF THE SPECIFICATIONS AND MATERIALS NOW. IF YOU HAVE ANY QUESTIONS ABOUT THESE TERMS, PLEASE CONTACT THE MANAGING DIRECTOR OF THE INTERNATIONAL PRESS TELECOMMUNICATION COUNCIL. AS OF THE DATE OF THIS REVISION OF THIS SPECIFICATIONS LICENSE AGREEMENT YOU MAY CONTACT THE IPTC at www.iptc.org.

License agreement version of: 30 January 2006

Contents

---

1 Introduction to EventsML-G2

EventsML-G2 specifications are a member of the Family of IPTC G2-Standards which is built on a common structural and function framework called the IPTC News Architecture (NAR). For this reason many components of EventsML-G2 are common with other members of the G2-Standards, e.g. NewsML-G2.

To better understand the terminology the IPTC used for the G2-Standards specification we recommend the **Glossary** (page 248) as reference, as it provides an extensive set of terms and their definitions.

1.1 Overview

1.1.1 What is EventsML-G2?

♦ EventsML-G2 is a standard for conveying event information in a news industry environment.

♦ EventsML-G2 is a member of the Family of IPTC G2-Standards, this family shares a lot of common specifications (the IPTC "News Architecture - NAR") for the exchange of news items and knowledge about topics and concepts.

♦ EventsML-G2 may be used for:

  ● Receiving all facts about an event from the event organiser,

  ● Publishing all facts about a specific event by a news provider,

  ● Publishing all or only a subset of the facts of one to many events by event listings,

  ● Adding information regarding the coverage of an event by a news provider to the distributed event facts, e.g. for daybooks,

  ● Storing facts about knowledgeable events in archives.

1.1.2 Business Advantages of Using EventsML-G2

EventsML-G2 are:

♦ Comprehensive (= many types of events may be covered).

♦ Flexible (= copies of substructures may be used many times, e.g. all the people appearing at an event).

♦ Extensible (= news provider specific data may be added) data structure to capture facts about events.

EventsML-G2 may express facts and information about events either by literal text (free text) or by codes from controlled vocabularies.

EventsML-G2 provides very flexible date types:

♦ year, month, day, optionally plus time

♦ year and month only or even year only

♦ approximative dates = a date range

EventsML-G2 reuses building blocks from the common News Architecture allowing to reuse software components, making their implementation cheaper.

EventsML-G2 makes use of industry standards: allows processing with standard tools. The EventsML-G2 syntax is built on XML, the Extensible Markup Language of the W3C, furthermore, EventsML-G2 makes use of W3C XML Schema and complies with the basic notion of the Semantic Web, the Resource Description Framework (RDF). This allows an easy transfer of EventsML-G2 structures to other XML-based standards and the integration of information about an event into the Semantic Web.

1.1.3 What is an Event – to be represented by EventsML-G2

An event is "something that happens" by definition. For the news industry, it is "something that happens and is subject to news coverage." All the events in a day make up an "agenda," which can be a marketable product sold to clients or simply an internal daybook used by editors to organise their work.

An event is planned or unplanned, with breaking news capable of overshadowing everything on the schedule.

Breaking news can generate a series of planned events; it becomes part of the daily news agenda the moment a decision is made to cover it. For agencies, this occurs with the first advisory announcing plans to provide coverage. For broadcasters it comes with the dispatch of a news team to the site; for newspapers it is when space is reserved for the story or page makeups rearranged.

Automated systems need to store and exchange information about news events. This is currently done in an ad-hoc manner, leading to overly-specialized formats and incompatible exchange by models. From that the IPTC learned that the industry would benefit from an event information interchange standard. Such a standard would facilitate the efficient exchange of event information, and the creation of better tools to support event management.

1.2 Definitions

1.2.1 Event Information

The event information describes a particular event in detail. This includes the "who", "what", "when", and "where" information for the event along with identification and publication (news management) information. The event information also includes facilities for relating events to each other and relating news items (both complete and incomplete) to the event information.

1.2.2 Coverage Information

The coverage information describes the plan of news coverage for this event. Each event may have zero or more assignments containing this coverage information. This information can be used internally within a news organisation for assignment of resources, planning of coverage, etc. It can also be used to publish information about expected coverage, so that consumers of the news coverage can plan their own news coverage accordingly.

1.2.3 The Data Model

The data model for EventsML-G2 has to cover two different facets of event information which relate to a basic distinction made for all G2 standards:

♦ Topical News: is typically volatile information in the sense of "nothing is older than yesterday's news".

♦ Persisting Knowledge: is information which is remembered and referenced to for a long time.

For EventsML-G2 this is reflected by two different data models:

♦ Volatile information about an event is represented by an "event" structure which is plugged into a NewsML-G2 news item as its content. A single news item may include one to many event structures. This kind of event information cannot be referenced as persisting information from any other item. Find details about this data model in section **Events in a NewsItem** (page 17).

♦ Persistent information about an event is represented by an EventsML-G2-Concept Item which is a generic NAR structure for concepts extended by a set of detailed information specific to an event. As any other kind of a Concept Item also this specific one for events can be referenced by its Concept Identifier.

The same applies to KnowledgeItems: a variant with event specific extensions is available, in particular event details are added to the concept structure inside the KnowledgeItem.

Find details about this data model in section **An Event in a Concept Item** (page 18).

The most important thing to note about the EventsML-G2 data model is that the core structures holding information about an event are identical for both the content plugged into a News Item and the extension of a Concept Item. Hence it is very easy to build a single EventsML-G2 processor for topical and persisting information about an event.

1.3 Conformance Levels

Another feature EventsML-G2 inherits from the NAR are the two conformance levels "Core" and "Power".

Different conformance levels are defined in the model, each of them related to a level of complexity (at the conceptual and processing level) of the related Items. This feature adds modularity to the model.

The current model defines two conformance levels named "core conformance level" (CCL) and "power conformance level" (PCL). The core conformance level is focused on simplicity and interoperability. The power conformance level is a superset of the core conformance level which gives more flexibility to providers who choose it, but the recipient processors are more complex to program to comply with PCL and interoperability is lower than for CCL as not all recipients will implemented the power level.

A compliant processor must therefore assert supporting either "core" or "power" functionality.

As the "power" features are only an extension of the "core" features, a "core" compliant processor SHOULD process "power" Items by simply ignoring the information pertaining to the "power" level.

EventsML-G2 specifies does not specify in its own scope data structures which are different for the two conformance levels, but it inherits specifications of datatypes of properties and attributes from the NAR which are different at the two conformance levels. The data types corresponding to the conformance level are defined in the specification tables in the **Specification Reference** (page 44).

1.4 EventsML-G2 and iCalendar

A well known and widely used standard for events data is "iCalendar" which is specified by RFC 2445.

EventsML-G2 compares very well to it as it covers virtually all features of an iCalendar Event Component:

Table 1. iCalendar-to-EventsML-G2 Component Mapping

iCalendar Event Component (Alphabetically)	Corresponding EventsML-G2 Component
attach	"link" property of a G2-item
attendee	"participant" property
categories	"subject" property
class	Access management functionality, no direct equivalence in EventsML-G2
comment	"note" property (under "event" for news and "concept" for a concept)
contact	"contactInfo" property (under eventDetails)
created	"contentCreated" property (in contentMeta) of a G2-item for news or a concept.
description	"definition" property (under "event" for news and "concept" for a concept)
dtend	"end" property (under eventDetails/dates)
dtstamp	"contentCreated" property (in contentMeta) of a G2-item for news or a concept.
dtstart	"start" property (under eventDetails/dates)
duration	"duration" property (under eventDetails/dates)
exdate	"exDate" property (under eventDetails/dates)
exrule	"exRule" property (under eventDetails/dates)
geo	"position" property (under eventDetails/location/geoAreaDetails)
last-mod	"contentModified" property (in contentMeta) of a G2-item for news or a concept.
location	"location" property (under eventDetails)

Table 1. iCalendar-to-EventsML-G2 Component Mapping (Continued)

iCalendar Event Component (Alphabetically)	Corresponding EventsML-G2 Component
organizer	"organiser" property (under eventDetails)
priority	As this iCalendar property reflects the priority for a calendar of an individual no equivalent exists in EventsML-G2.
rdate	"rDate" property (under eventDetails/dates)
recurid	No direct equivalence in EventsML-G2, assigned functionality may be replicated by G2-item means.
related	No direct equivalence, but relationships can be expressed by other G2-item means
resources	Not covered by EventsML-G2 1.0, planned for future versions.
rrule	"rRule" property (under eventDetails/dates)
rstatus	Scheduling protocol functionality is not covered by EventsML-G2
seq	"version" attribute of the G2-item's root element
status	"confirmation" (under eventDetails/dates) reflects the status of confirmation of the dates of the event, while "occurStatus" (under eventDetails) reflects the overall status of the event.
summary	"name" property (under "event" for news and "concept" for a concept)
transp	Not covered by EventsML-G2
uid	"guid" attribute of the G2-item's root element
url	No direct equivalence. For G2-items it may be defined individually by each news provider how to resolve the identifier of an G2-item to an accessible location.
x-prop	EventsML-G2 provides "Extension points" for this purpose.

2 Events

2.1 The Core Information about Events

Regardless whether the information about an event is topical or persistent (see **The Data Model** on page 13) the same structure is used to mark it up.

The information about an event includes, first a set of more generic properties:

♦ A natural language **name** (page 70) for the event. This name should be rather concise and could be expressed in different languages.

♦ A natural language **definition** (page 67) for the event and it could be more extensive than the name, it could explain facets in detail. It can also be expressed in different languages.

The *role* attribute of a definition could be used to provide variants of an explanation, e.g. a short one for overviews and a rather extensive one for a detailed presentation.

♦ A natural language **note** (page 153) about the event. This could be an explanation of details or background information regarding the definition. Again this note can be expressed in different languages and can be qualified by a *role* attribute.

♦ A **facet** (page 112) property to further qualify the nature of the event. The facet can take either literal values or values from a controlled vocabulary and could be used to express e.g. that this event is a concert, a hockey game or a press conference.

♦ The properties **sameAs** (page 186), **broader** (page 59), **narrower** (page 149) and **related** (page 175) can be used to define a relationship with this event to another event or concept.

In particular broader may be used to express that this event is a sub-event to another event, e.g. a break-out session of a big conference, one competition of the Olympic Games or one of the concerts of a festival.

Then a set of rather event-specific properties - wrapped by the **eventDetails** (page 106) property:

♦ A **dates** (page 97) sub-structure to express the start date and the end date or duration of the event. This includes using the "approximative dates", i.e. a range of dates, and on date in this range as a kind of best guess or most likely date.

If this event is recurring this can be expressed by means of recurrence properties which align to equivalent properties of the iCalendar standard RFV 2445, see more below.

♦ An **occurStatus** (page 154) to indicate the status of the occurrence - if this is a unplanned or planned event, and if it is planned how likely it is to occur.

♦ A set of **registration** (page 174) information which may be used to define how persons have register for the event, this may include the accreditation of journalists.

♦ A set of **accessStatus** (page 48) information.

♦ A set of **participationRequirement** (page 161) properties. This could be used e.g. for expressing age limits - think of required parental guidance for movies - or for formal requirements for training course events.

♦ A set of **subject** (page 193) properties to express what the event is about. Be aware of the difference between a facet and a subject: a facet should indicate the nature of the event, what the event is, while a subject indicates applicable categories for what the event is about. For example, "concert" is a facet, while "music" or "Wolfgang Amadeus Mozart" is a matching subject.

♦ A set of **location** (page 145) properties. In most cases it will be the only location of where the event will take place - but e.g. festivals could have more than one location.

♦ A set of **participant** (page 160) properties to list all kinds of parties appearing in different roles at the event - the particular role can be expressed by the *role* attribute.

♦ A set of **organiser** (page 158) properties to list all parties involved in organising the event - the particular role can be expressed again by the *role* attribute

♦ A set of **contactInfo** (page 73) properties for the event. Be aware that the location, the participant and the organiser properties may contain contactInfo structures, but they pertain only to this particular property while this contactInfo is to be used for the event as a whole.

♦ A set of **language** (page 140) properties to reflect all languages which will be spoken at the event.

♦ A set of **newsCoverage** (page 150) properties. They can provide all necessary information about the planned coverage for this event by a news provider. To express this planned coverage either a free-text property can be used or provider specific additional properties.

♦ As for many wrapping elements in G2-Standards, the information about an event can also be extended by provider-specific properties.

2.2 Event Information in Items

2.2.1 Identification and Versioning of Items

It is possible to positively identify any kind of an item specified by the G2-Standards as it moves through the news workflow, and is transferred from place to place and from system to system.

Each and every item – including News Items, Concept Items, Knowledge Items and Package Items – MUST have a *guid* attribute, that is a persistent and globally unique identifier. The guid is required to be in the form of an IRI. Any IRI capable of acting as a globally unique identifier is accepted.

Note: the IPTC will provide the newsml-URN for this purpose, specified by a successor of RFC-3085.

A newsItem MAY have a *version* attribute, and this version MUST be incremented when the content of the Item is updated. The first version MUST be numbered 1: if the version is not explicitly set, the value "1" must be assumed as default by the recipient of the Item.

The *standard* attribute must reflect the G2-Standard which governs the structure:

♦ For topical events in a News Item this is NewsML-G2.

♦ For persisting information about an event in a Concept Item this is EventsML-G2.

The *standardversion* attribute must reflect the version of the standard as it is implemented by the corresponding XML Schema.

Samples:

```
<newsItem standard="NewsML-G2" standardversion="2.0"
    guid="urn:newsml:iptc.org:20071231:newssample" version="2"
    xmlns="http://iptc.org/std/nar/2006-10-01/" >
</newsItem>

<conceptItem standard="EventsML-G2" standardversion="1.0"
    guid="urn:newsml:iptc.org:20071231:eventsample" version="4"
    xmlns="http://iptc.org/std/nar/2006-10-01/" >
</conceptItem>
```

2.2.2 Events in a NewsItem

Topical event information must be conveyed by using the NewsML-G2 NewsItem (see **Representing News - newsItem** on page 20) as a wrapping item instance. The structure of a NewsItem defines a special node where to attach content plug-ins, the inlineXML element.

For EventsML-G2 an **Events Wrapper** (page 108) element will be added as child to **Inline XML (NewsML-G2 Specific)** (page 129) and it acts as a wrapper of one-to-many **Event** (page 105) elements, each representing the topical information of a single event.

Figure 1. Event Information in a News Item
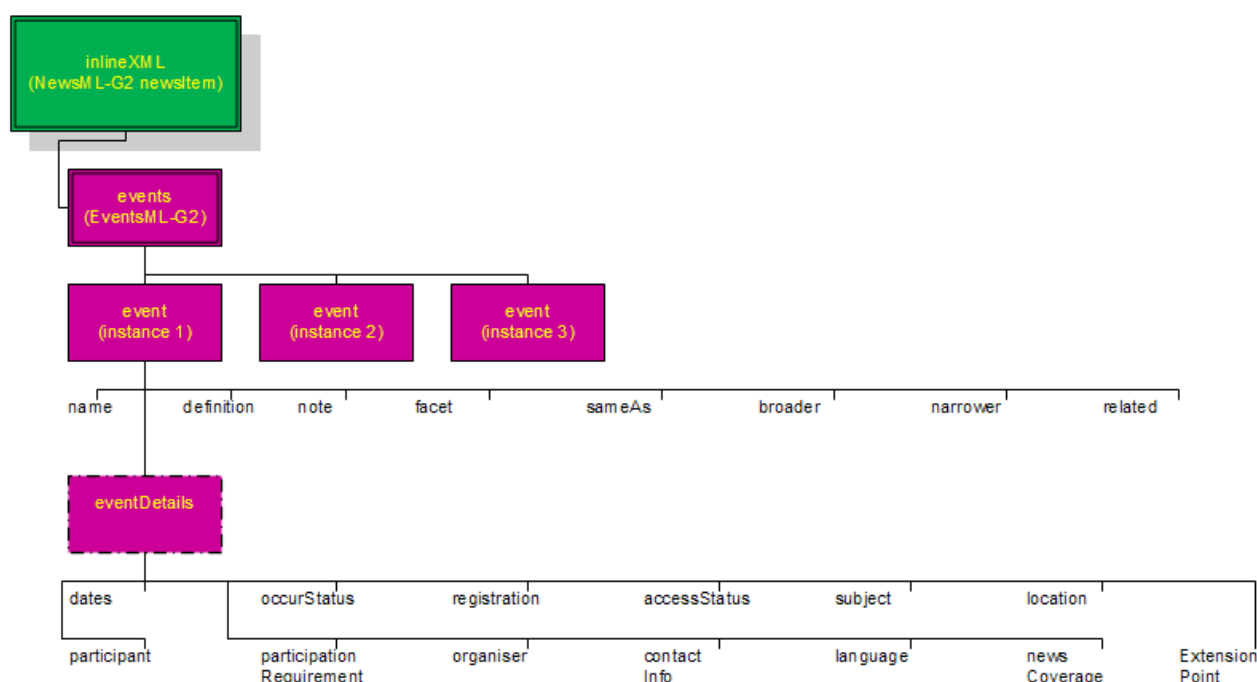
The event element wraps a group of more generic descriptions and a couple of details about an event. The first group is made of a short name which can be displayed as a one-liner, a more comprehensive definition of the event and a note with supplemental information.

A sibling to this group is eventsDetails, it wraps all the details of the event, when and where it happens, who is involved and how to get there.

Finally optional information about the planned news coverage of this item may be added.

2.2.2.1 News Metadata

In general the News Metadata section of a NewsItem - wrapped by the **Content Metadata for News Items** (page 76) element - should be populated and used as specified for NewsML-G2.

Further to this general recommendation these event specific considerations apply:

♦ If more than a single event is conveyed by a NewsItem the content metadata apply to the set of events as a whole. In most cases this set will be selected from a larger repository by some rules, like "events of next week", or "music events". This could be reflected by e.g. the headline, the description or even the subject property.

♦ Genre property: an appropriate value should be applied, like "almanac" or "daybook" from the IPTC Genre NewsCodes

♦ Language property: be aware of the difference between the language property of the content meta-data - it reflects the languages used in the content, in this case in the description of the events - and the language property of the event structure - it reflects a language which is used at an event.

2.2.3 An Event in a Concept Item

The persisting knowledge facet of event information is represented by the EventsML-G2 Concept Item, an extended variant of the generic G2 Concept Item (see **Managing Individual Concepts - conceptItem** on page 33), and subsequently the EventsML-G2 Knowledge Item, an extended variant of the generic G2 Knowledge Item (see **Managing Sets of Concepts - knowledgeItem** on page 34).

Any Concept Item or Knowledge Item provides a group of generic definitions and a set of details specific to a kind of concept, in this case specific to an event.

Generic definitions are part of the basic Concept Item or Knowledge Item and include:

♦ The Concept Identifier for this event.

♦ Relationships to other events.

♦ A name, a definition, explanatory notes and refining facets.

Find more about the generic part in section **Representing Concept Information - concept Component** (page 31).

In EventsML-G2 Concept Items the value of the type of a concept (conceptItem/concept/type) must be set to concept URI of http://cv.iptc.org/newscodes/cpnature/event which may translate to a QCode of cpnat:event.

Figure 2. Event Information in a concept element
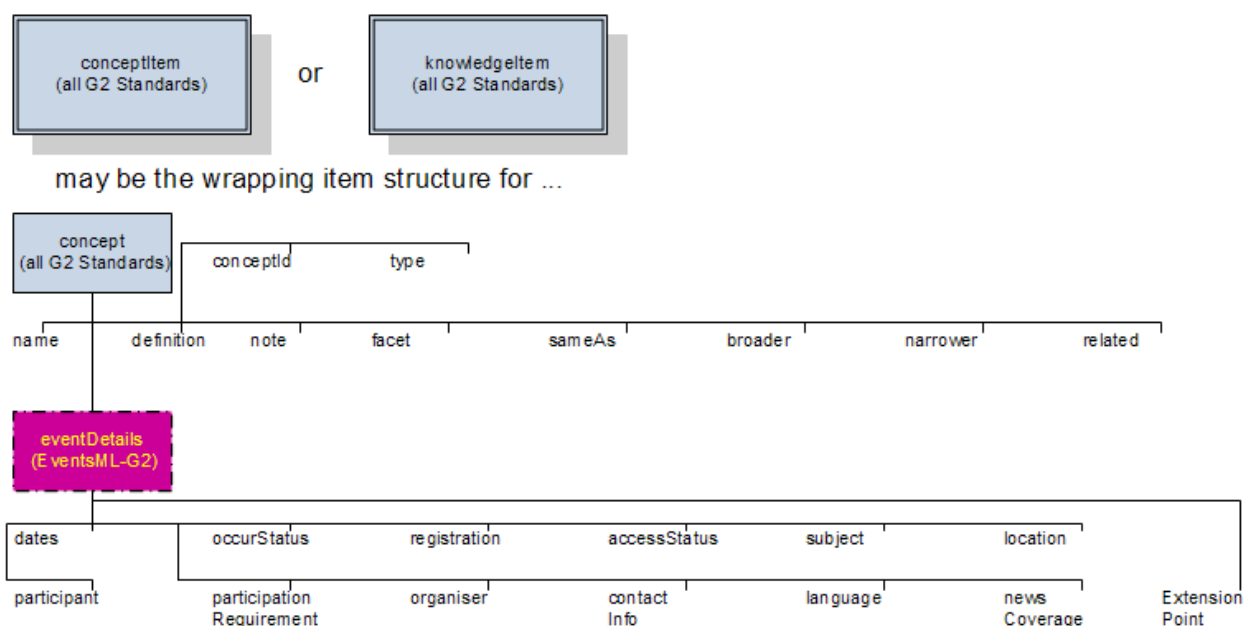
The event specific details are expressed by an eventDetails structure plugged into the "concept" of a Concept Item or a Knowledge Item. The eventDetails used there are completely identical to the structure with the same name used for the "event" element in the content set of a News Item.

The Concept Identifier of an event may be used by other items (either News Items or Concept Items) to reference this event. On a purely technical level this Concept Identifier can be used for any "qcode" qualifier of a property. On a semantic level the only prerequisite are reasonable semantics of the property to reference an event – e.g. a property not limited to persons or locations by its semantics.

Examples are:

♦ Using an event's Concept Identifier as QCode for the "subject" property of a News Item. This indicates that the content of the News Item is about this event, the News Item's content may be a text, a photo, audio or video covering the event.

♦ Using an event's Concept Identifier with the "sameAs", "broader", "narrower" and "related" properties of another Concept Item. By these means a structure or network of events can be created, e.g. to link individual performances with a cultural festival or different talks to a conference.

3 Representing News - newsItem

An XML Schema file corresponding to the specifications for this item is available (see **The Full Set of Specification Documents** on page 3).

3.1 Description

A newsItem aims to convey news with the sense of the reporting of a newsworthy event or fact. Its content is gathered by journalists, presented with a journalistic style, and updated according to the progression of the story.

Examples of newsItems are a news report, a picture, a graphical illustration of some event, a video clip or an illustrated biography.

Typical characteristics of a newsItem are:

- ♦ Its content may be of any media type or format, e.g., the thumbnail, preview and high definition renditions of a picture.

- ♦ It can also convey more structured news information, e.g., information about companies, sports events and general events, in instances when this information is related to an event or fact.

- ♦ Its content is of short term interest: newsItems are volatile, and interest in them fades as time passes ("nothing is older than yesterday's news").

- ♦ It is expressed via a set of alternative renditions of some media content.

- ♦ It will usually be updated only for a short period of time, as long as the covered event evolves, and then may be archived.

- ♦ It refers to an arbitrary set of concepts and entities.

- ♦ It may be associated with other newsItems or Web resources via typed links.

3.2 Indication of Compliance with a Standard and Conformance Level

The IPTC newsItem *standard* attribute MUST be set to "NewsML-G2".

The schema version to which the newsItem conforms MUST be indicated as an attribute. The current version is identified by the string "2.0".

The IPTC conformance level to which the newsItem conforms in this specification MUST be indicated by the value "power".

Sample:

```
<newsItem standard="NewsML-G2" standardversion="2.0"
     xmlns="http://iptc.org/std/nar/2006-10-01/" >
</newsItem>
```

3.3 Identification and Versioning

It is possible to positively identify a newsItem as it moves through the news workflow and is transferred from place to place and from system to system.

A newsItem MUST have a *guid* attribute, which is a persistent and globally unique identifier. The guid is required to be in the form of an IRI. Any IRI capable of acting as a globally unique identifier is accepted.

Note: The IPTC will provide the newsml-URN for this purpose, specified by a successor of RFC-3085.

A newsItem MAY have a *version* attribute, and this version MUST be incremented when the content of the Item is updated. The first version MUST be numbered 1: if the version is not explicitly set, this value must be assumed by the recipient of the Item.

The *standardversion* attribute must reflect the version of the standard as it is implemented by the corresponding XML Schema.

Sample:

```
<newsItem standard="NewsML-G2" standardversion="2.0"
      guid="urn:newsml:iptc.org:20071231:sample" version="2"
      xmlns="http://iptc.org/std/nar/2006-10-01/" >
</newsItem>
```

3.4 Catalog of Controlled Vocabularies

NewsML-G2 recommends the use of controlled values for most properties. Each news provider is free to use their own taxonomies of subjects, genres, geopolitical areas, organisations etc., and to use any value scheme it decides in the Items it provides. A provider must therefore indicate the list of the schemes he is using.

Cataloguing information MUST be included at the top of each Item.

A catalog is defined as a set of scheme declarations in use by a news provider for a given Item.

Due to the large number of schemes potentially used in a single Item, and knowing that bandwidth is very important to the News industry, the catalog may be stored remotely.

A remoteCatalog MUST have a *href* attribute which contains the URI of a remote catalog. A remote catalog takes the form of an XML file with a catalog element as root.

The URI of a remote catalog acts both as a locator and a global identifier, therefore:

- ♦ The URI of a remote catalog MUST NOT be relative.
- ♦ If a remote catalog is functionally changed, the IRI used to access it MUST be changed. Functional changes are:
  - the addition or removal of a scheme declaration,
  - a change to a scheme alias,
  - a change to a scheme URI.

One or more additional title for a catalog or catalogRef MAY be provided in different languages and variants.

As some required properties take a QCode as a value, at least one catalog or remoteCatalog MUST be present.

In general, a given provider will define a unique catalog of all used schemes, store it in a central repository and reference it from all Items it provides. A provider MAY declare several catalogs in the same Item. This may be especially useful for an aggregator which uses property values from different sources, but requires a way to avoid scheme alias clashes. In this case, catalog and catalogRef elements MAY appear in any order, and their order is not relevant.

Detailed information on the structure of catalogs and their processing is given in **Dealing with Controlled Values** (page 37).

Sample:

```
<newsItem  standard="NewsML-G2" standardversion="2.0"
      guid="urn:newsml:iptc.org:20071231:sample" version="2" xmlns="http://
      iptc.org/std/nar/2006-10-01/" >
      <catalogRef href="http://aprovider.com/cv/newsml-g2-catalog-4.xml"/>
      ....
</newsItem>
```

3.5 Signature Information

A digital signature may be associated with a whole Item or only parts of it. For example, it is possible to sign each individual news content component of a newsItem using their local identifiers as a local reference.

A digital signature is a unique seal placed on data. It is very difficult to forge and assures that any change made to the signed data cannot go undetected.

This specification supports the model and syntax defined by the W3C in [XMLDSIG], and introduced by the following: "XML Signatures provide integrity, message authentication, and/or signer authentication services for data of any type, whether located within the XML that includes the signature or elsewhere".

This specification model excludes two functionalities defined by the W3C XML-Signature Processing Recommendation. These are: "Signed content included within an XML Signature Construct" and "Detached Signatures".

Therefore this specification offers the following features:

♦ A Signature MUST be "enveloped" (the Signature Component is contained within the Item being signed).

♦ A Signature MUST sign the Item containing the Signature component or child components of the Item containing the Signature.

♦ The Signature MUST NOT be "enveloping" (it cannot sign content found within the signature itself).

♦ A Signature MUST NOT be "detached" (a detached Signature Component would not be contained within the Item being signed and could be external to the containing document).

♦ A Signature MUST NOT be related to Items and Components external to the enclosing document (via references).

3.6 Rights Information

The content of a newsItem is bound to a set of copyrights and licensing information.

A rightsInfo wrapper element acts as a container for a set of properties related to rights, which offer a basic expression of the copyright and usage conditions associated with an Item.

This set is limited to an accountable person, a copyrightHolder and a set of copyrightNotices and usage-Terms.

The order of the properties is flexible: The non-repeatable properties MUST come first, then the repeatable properties MAY be inserted in any order.

The expression of rights can be verbose, and the volume of information exchanged or stored may suffer from the repetition of such information. Therefore each property provides an *href* attribute as an alternative locator of a remote expression of rights. In the case where both inline and remote expression of rights is indicated, the inline expression MUST take precedence.

In some situations, different parts of the content are associated with different sets of rights; the rightsInfo element is therefore repeatable.

Each set of rights provides an optional reference attribute (idrefs) which indicates which part of content is bound to these rights. This can be achieved by referencing the id of a content component of an item, i.e. inlineXML, inlineData and remoteContent for a News Item, and concept for a Knowledge Item. The idrefs attribute MAY reference more than one component.

A same piece of content MAY have different set of rights associated with it; in this case the union of the rights applies to this content. In case of a logical clash, the recipient SHOULD consider it as an error and check with the provider.

The rightsInfo element also provides optional time validity attributes (*validfrom* and *validto*) which express the date and time between which the set of rights properties apply.

Each provider may add a set of metadata properties which have to be defined in a non-IPTC namespace. See also **XML Namespaces** (page 44) and **Extension Points in XML** (page 45).

3.7 Item Metadata

Such information is wrapped in the itemMeta wrapper element and split between news management metadata and Item links.

3.7.1 Management Metadata

Management metadata is bound to the Item as a whole and reflects its processing in a professional work-flow.

The order of the properties in this set is imposed by the W3C XML schema.

Table 2. Item Management Group Elements

Element Title	Element Name	Card	Described on Page
Item Class	**itemClass**	(1)	134
Content Provider	**provider**	(1)	78
Date Item Version Created	**versionCreated**	(1)	90
Date Item First Created	**firstCreated**	(0..1)	89
Date Item Embargo Ends	**embargoed**	(0..1)	88
Publish Status	**pubStatus**	(0..1)	171
Role in the Workflow	**role**	(0..1)	181
File Name	**filename**	(0..1)	114
Generator Tool	**generator**	(0..1)	115
Profile	**profile**	(0..1)	169
Editorial Service	**service**	(0..unbounded)	102
Item Title	**title**	(0..unbounded)	138
Editorial Note	**edNote**	(0..unbounded)	101
Member Of	**memberOf**	(0..unbounded)	147
Instance Of	**instanceOf**	(0..unbounded)	131
Signal	**signal**	(0..unbounded)	189
Alternative Representation	**altRep**	(0..unbounded)	53

The IPTC provides a standardised scheme applicable to the itemClass property of a newsItem, identified by the URI http://cv.iptc.org/newscodes/ninature/.

Each provider may add a set of metadata properties which have to be defined in a non-IPTC namespace. See also **XML Namespaces** (page 44) and **Extension Points in XML** (page 45).

3.7.2 Processing the Publish Status of an Item

The IPTC makes these values normative for the exchange of Items between a provider and its customers:

- ♦ Usable: The Item MAY be published without restriction.

- ♦ Withheld: Until further notice, the Item MUST NOT be published or used under any circumstances. If the Item has been published the publisher MUST take immediate action to withdraw or retract it.

- ♦ Canceled: The Item MUST NOT be published or used under any circumstances. If the Item has been published the publisher MUST take immediate action to withdraw or retract it.

Embargoes are managed by the embargoed property, which indicates the date of end of an embargo. This property is optional; if present, the date and time it contains must be compared with the current date and time before the Item is used. The Item must not be published before this time. The embargo over-rides the usability of the Item, inferred from its status. If embargoed does not exist the item is not embargoed.

3.7.2.1 State Transition Diagram

This depicts the state transition diagram reflecting the ways in which the pubStatus values are intended to be used. Thus, upon creation of an Item, allowed statuses are usable and withheld. It is possible to withhold a usable document; it is possible to release a withheld document; it is possible to cancel a usable or withheld document. Once an Item has had its status set to canceled, it has reached a final state.

Figure 3. State Transition Diagram
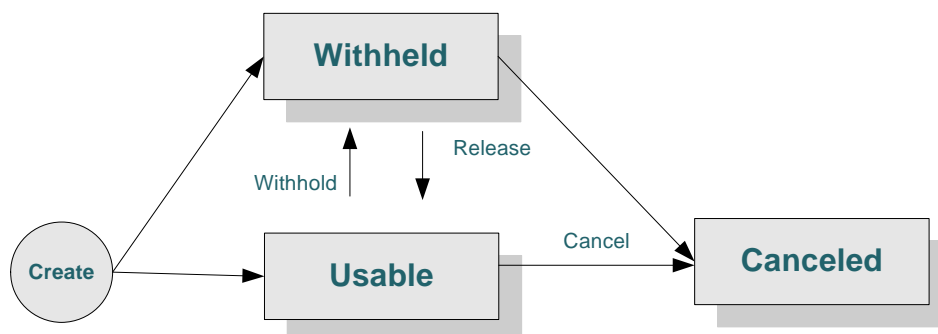
3.7.2.2 Use Cases Associated with a Status of Withheld

Use Case 1: A provider distributes a story as a newsItem (version 1) with the status usable. At a later stage he learns that there may be a problem with the information included in the Item. He sends a new version of the newsItem (version 2) with a status set to withheld. All recipients systems must display a warning on this newsItem, and recipient publishers must postpone the publication of the information contained in the newsItem until further notice. The news provider has confirmation that the information is false and decides to set the status to canceled (version 3).

Use Case 2: An eCommerce system proposes a large collection of illustrated articles managed as news items. The publisher managing the system sees that the information included in a newsItem (version 1) is not up to date anymore, and decides to hide this Item from its customers until it is properly revised. He set then its status to withheld (version 2), edits the newsItem and set its status back to usable (version 3).

3.7.2.3 Processing Model on the Recipient Side

Here is the processing model on the recipient side and relies on the pubStatus and embargoed properties:

```
Test pubStatus = canceled:
     The Item must not be used, ever. Any usage of the Item must be prohib-
     ited, if needed by the way of alerts.
     Else: next
Test pubStatus = withheld:
     The Item must not be used until further notice. Any usage of the Item
     must be prohibited, if needed by the way of alerts.
     Else: next
Test pubStatus = usable:
     Test embargoed is not past: if yes, the live status is embargoed
     Else: The Item is usable and may be safely published
```

3.7.3 Processing of versionCreated

If the value provided by any date/time field does not conform to the appropriate syntax (e.g. format "YYYY-MM-DDTHH:MM:SS[+-]MM:SS") it MUST be considered as being not existent.

In the case of the mandatory versionCreated property the full Item MUST be considered as being void.

3.8 Item Links

A powerful feature of NewsML-G2 is the capability to associate Items via links. It is therefore possible to create a network of news resources, for management and navigation purposes.

The link element offers a generic mechanism for linking Items within the NAR framework as well as creating links from Items to other Web resources.

The semantic of the link MAY be refined via a relationship attribute (*rel*). In the absence of such indicator, the implied meaning of the link is "see also" (i.e. a navigation link).

The IPTC provides a recommended scheme of link relationships identified by the URI http://cv.iptc.org/newscodes/relation/.

If the target resource is an Item, the *guidref* attribute SHOULD identify the target Item by its globally unique identifier and a *version* attribute MAY indicate the target Item version; in the absence of version information, the target resource is the latest version available. The *href* attribute MAY additionally indicate the location of the target resource.

If the target resource is a Web resource, the *href* attribute MUST indicate the locator of the target resource.

The content type, a.k.a. IANA MIME type of the target resource MAY also be indicated by a *hreftype* attribute.

In order to ease the processing of a link, the size of the target resource MAY be added as an attribute. Such feature is especially useful if the target on the link is a Web resource. If the target resource is an Item, the size which is given here MUST be the size of the XML representation of the Item.

A *rank* attribute may represent the rank of the link among other links.

This property also provides time validity attributes (*validfrom* and *validto*) which express the date and time between which the link is valid.

A title child element describing the link MAY be added for display to the users.

Supplemental metadata extracted from the target resource (usually an Item) may be added to the linking information as child elements. Such information is not constrained by the model. It may be part of the target Item Metadata (e.g. Publish Status, Alternative Location ...), Content Metadata (e.g. Intended Audience, Subject, Genre …) or Characteristics of the content (e.g. Size, Content Type, Format, or specific characteristics like the Height and Width of a picture). Different sets of characteristics may be provided, corresponding to specialized content components.

All properties SHOULD be included directly under the link property, without wrapper element (e.g. neither itemMeta nor contentMeta).

3.8.1 Processing Links

Link processing rules:

Link.1: Processor on the consumer side: If a guid and a version are provided, check whether the specific version of the Item is accessible using this information.

Processor on the provider side: If a guid and a version are provided deliver only the item version with the requested version number.

Link.2: Processor on the consumer side: If only a guid is available and no version, check whether an item is delivered by the provider. Consider a delivered version of the item as being the latest one.

Processor on the provider side: if only a guid is requested and not version, check if any version of the item exists, and if yes provide the one with the highest version number.

Link.3: Check whether the value of the *href* attribute allows some direct retrieval of the target resource via the Web (e.g. if the scheme is http: or ftp:), or an implicit resolution mechanism (e.g. DOI).

Link.4: Check whether an Alternative Representation (altRep) is exposed in the link. This information may complement the *href* attribute and provide an immediate URI resolution mechanism for Items. Multiple locations may be given, as allowed in the Item Metadata component. In such a case the processor will use the role qualifier and URL scheme for choosing the most appropriate resource.

Link.5: Signal an error or ignore the link.

3.9 News Content Metadata

News Content Metadata is directly associated with the news information conveyed by the Item, independently of the processing of the Item in a professional workflow. Such information is wrapped in the contentMeta wrapper element and split between administrative and descriptive metadata.

3.9.1 Administrative Metadata

This is a set of properties associated with the administrative facet of content, i.e. data that cannot be inferred from "consuming" (reading, listening to, watching) the content.

All properties are optional. The order of the properties in this set is flexible: the non-repeatable properties MUST come first and then the repeatable properties may be inserted in any order.

Table 3. Administrative Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Urgency	**urgency**	(0..1)	198
Date Content Created	**contentCreated**	(0..1)	86
Date Content Modified	**contentModified**	(0..1)	87
Located	**located**	(0..unbounded)	144
Information Source	**infoSource**	(0..unbounded)	133
Creator	**creator**	(0..unbounded)	85
Contributor	**contributor**	(0..unbounded)	80
Audience	**audience**	(0..unbounded)	56
Excluded Audience	**exclAudience**	(0..unbounded)	109
Alternative Identifier	**altId**	(0..unbounded)	52

3.9.1.1 Dates Processing Model

Two optional dates are associated with the content of an Item.

contentCreated and contentModified processing rules:

DatesGeneral.1: If the value provided by any date/time field does not conform to the appropriate syntax (e.g. format "YYYY-MM-DDTHH:MM:SS[+-]MM:SS") it MUST be considered as being not existent.

DateValues.1: If contentCreated is present it MUST NOT be later than versionCreated.

Error handling if it is later: at the creator's site an error alert should be issued, on the receiver's site it should be set to versionCreated.

DateValues.2: If contentModified is present contentCreated SHOULD be present as well.

In this case contentModified MUST NOT be earlier than contentCreated.

Error handling if it is earlier: at the creator's site an error alert should be issued, on the receiver's site it should be set to contentCreated

DateValues.3: If contentModified is present it MUST NOT be later than versionCreated.

Error handling if it is later: at the creator's site an error alert should be issued, on the receiver's site it should be set to versionCreated.

DateProcessing.1: The recipient processor MUST first check if a contentModified element is present.

DateProcessing.2: If not it MUST check if a contentCreated element is present.

DateProcessing.3: If not it SHOULD assume that the content was created at the time indicated by versionCreated element in itemMeta.

3.9.1.2 Audience Processing Model

Audience processing may be used to form ad hoc groups of recipients for which the Item is particularly significant or to filter out some users from the list of intended recipients of an Item.

The audience is expressed as a set of "positive" values (audience) and a set of "negative" values (exclAudience). The logic is to make the content easy to find to the audience identified by the positive values, but keep this content away from the audience identified by the negative values. An attribute of each property may indicate the expected significance of the content for this specific audience, and acts as a threshold for recipient filters.

The model for the audience processing is only a part of the overall filter that is used to determine whether a particular recipient is entitled to have access to the Item. It could be combined with the processing of other properties to further narrow the number of Items that match the recipient profile.

The processing rule has to be considered as a function which returns TRUE to indicate the recipient is entitled to receive the content, FALSE in case he is not entitled and NULL if the item does not contain any audience statements that apply to the Recipient.

Audience processing rules:

Audience.1: If any of the exclAudience properties applies to the recipient: return FALSE

Audience.2: If any of the audience properties applies to the recipient: return TRUE.

Audience.3: Return NULL.

3.9.2 Descriptive Metadata

This is a set of properties associated with the descriptive facet of news content, i.e. data that can be inferred from "consuming" (reading, listening to, watching) the news.

All properties are optional, repeatable and may be inserted in any order.

Table 4. Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Language	**language**	(0..unbounded)	140
Genre	**genre**	(0..unbounded)	116
Subject	**subject**	(0..unbounded)	193
Slugline	**slugline**	(0..unbounded)	190
Headline	**headline**	(0..unbounded)	122
Dateline	**dateline**	(0..unbounded)	96
By	**by**	(0..unbounded)	60
Description	**description**	(0..unbounded)	98

3.9.3 Other Content Metadata

Each provider may add a set of metadata properties which have to be defined in a non-IPTC namespace. See also **XML Namespaces** (page 44) and **Extension Points in XML** (page 45).

3.10 Part Metadata

Streamed content may be split into different sections (called "shots" in the video world). Images may also be split in regions.

A specific set of metadata MAY be associated with any individual content part. Such metadata is wrapped in a partMeta element, which is repeatable in the newsItem and MUST be inserted after contentMeta.

Each part MAY have a part identifier (partId) and a sequence number (seq).

Each part MAY be illustrated by an icon - e.g. a keyframe of a video clip - which takes the form of an IRI. It is not mandatory for such icon to be a pure extraction of the content.

A section of a stream MAY be defined by a time delimiter (timeDelim). The time scope is expressed as *start* and *end* timestamp attributes plus an additional time unit (*timeunit*) attribute.

A region of an image MAY be defined by a region delimiter (*regionDelim*). In the current release, regions are limited to rectangles defined by {x, y, width, height} coordinates in pixels expressed as a set of attributes.

If, during the processing of the content, it appears that part delimiters do not correspond to any physical content, then the corresponding set of metadata MUST simply be discarded.

News Descriptive Metadata may be applied to each part, in complement to the descriptive metadata applicable to the whole content.

Each provider may add a set of metadata properties which have to be defined in a non-IPTC namespace. See also **XML Namespaces** (page 44) and **Extension Points in XML** (page 45).

Note that partMeta is specific to the newsItem structure, and this feature is not present in other classes of Items.

3.10.1 Edit Units and Time Codes

It is recommended that time and durations are expressed in "edit units" (EditUnit), which represent the smallest editable portion of content, i.e. a video frame or an audio sample.

EditUnit = 1 / EditRate.

For video, the EditRate is the FrameRate.

For audio, the EditRate is the SampleRate.

The use of EditUnit is independent of the mode of representation of time (e.g. timecode) in editing devices. The timecode associates one value to each video frame or audio sample.

For video, the usual timecode format is HH:MM:SS:FF (Hours:Minutes:Seconds:Frames).

In the case of simple frame rates (e.g. 25 fps, 30 fps, 50 fps or 60 fps), the conversion of a number of EditUnits to timecode is simple.

However, there exist other frame rates (e.g. 29.97 fps, 59.94fps) for which this calculation requires more attention. A precise calculation would consist of replacing e.g. 29.97 fps by its exact value 1.001/30 fps and multiplying the number of EditUnits by 1.001 before conversion on the basis of 30 fps. Another method consists of calculating the timecode using the drop frame method defined in SMPTE 12M. The drop frame method is an approximation based e.g. on 29.97 fps (1.001001001/30 fps). The drop frame timecode is not systematically used, particularly if content is of a short duration with insignificant drift with the actual clock time. SMPTE 12M will evolve as it doesn't address higher frame rates with progressive scanning.

For audio, the usual video timecode (HH:MM:SS:FF) is used if the content also contains video. A time restricted timecode (HH:MM:SS) is often used for audio only content, although it can be completed by a number of complementary frames or its conversion in hundredths or thousandths of a second.

The time reference will be the one of reception or edition in the production system, which should be able to locate content in time based on the number of EditUnits.

3.10.2 Assertions About Concepts

When a concept is used as the value of many properties, it may be useful to group supplemental information about this concept at a unique location.

The optional and repeatable assert property provides information about a concept identified by a qualified code. The information is given as a set of properties relative to the concept. Many assertions may be included in an Item.

Any property of the concept may be included at this point, especially its name, its relationships with other concepts, its definition.

Note: This information is only up to date at the time of last modification of the Item.

3.11 References to Inline Concepts

When the same concept appears as a string in several different labels or in the textual content of a newsItem, it may be useful to group information about this concept at a unique location.

The optional and repeatable inlineRef property provides information about a concept found in some textual content. The string associated with the concept can be tagged by any element which provides an attribute of type ID. One or more local identifiers MAY be listed as value of the *idrefs* attribute of the inlineRef element.

The concept MUST be identified either by a qualified code or a literal value, and supplemental information MAY be given as a set of properties relative to the concept.

It is possible to offer alternative identifications of a concept if the concept extraction leads to ambiguous results. In this case several inlineRef properties will use the same local identifier.

It is possible to give values for the confidence with which the metadata has been assigned, the relevance of the metadata to the string to which it is attached, and why the metadata has been included.

3.12 newsItem Content

Content may be included by value or by reference, and useful characteristics are represented along with such content, in order to facilitate its processing.

Alternative renditions of the news content, i.e. different technical representation of the same logical content, are wrapped by a contentSet wrapper element. Their order of appearance in contentSet is of no relevance. Their presence is optional: this allows for a lightweight and progressive representation of information.

Each rendition SHOULD by defined by a *rendition* attribute.

All alternative renditions SHOULD be derived from an original rendition by a software processor. For example: images in different resolutions, vector graphics and alternative bitmap images, text in different formats (e.g. NITF and PDF). The rendition from which all other renditions originate is indicated by the *original* attribute of contentSet; this attribute takes as a value the local identifier (id) of the original content component included in the contentSet.

They are three kinds of content components, inline XML, inline data and remote content:

♦ The inlineXML wrapper element holds XML content which is directly embedded in the element. The root element of this structure must be the root element of the language. Content may belong to any XML language capable of expressing generic or specialized news information, e.g. NITF, XHTML, SportsML or XBRL. The XML vocabulary is identified by a content type attribute (*contenttype*).

♦ The inlineData wrapper element holds plain-text or base64 encoded content. Plain text or CDATA content MUST be identified by the "text/plain" content type. Binary content, like images, audio clips or even PDF or Word documents may be exchanged after proper encoding, but it is strongly recommended to use this structure for small assets only. The encoding algorithm MAY be indicated using the *encoding* attribute. In the absence of this attribute, the content must be plain text, and the con-

tent type must be setaccordingly. Encoding is not constrained to base64 at this level of conformance.

♦ The remoteContent wrapper element may be used for representing any kind of media and data format. The data is stored independently of the newsItem and is referenced via a hyperlink (href). The size in bytes of the remote content MAY be indicated. The element MAY also have time validity attributes (*validfrom* and *validto*) which express the date and time between which the reference is active.The same rendition of content MAY be present at different remote locations. In such a case the same value of the *rendition* attribute MAY be given to several remoteContent elements.

The description of the content in each content component MAY be complemented by a content type (contenttype), a format acting as an optional refinement of the content type, an indication on the software tool used to generate the content (generator) and the date and time when the content was generated, plus additional news content characteristics.

3.13 News Content Characteristics

News Content Characteristics are these physical properties of media content like the height and width of a picture, the word count of a news story or the duration of an audio clip, that help making selections among alternate renditions of news content.

The characteristics defined by the IPTC are a small and typical set of properties. Individual providers may add more characteristics they consider reasonable, i.e. audio data for professional broadcasting may require a different set from audio content for a podcast.

3.14 Channels

Some binary streams support the notion of channel or track: this is e.g. the case for DVD's, which are MPEG-2 encoded and provide several audio tracks in different languages. It may be important to indicate media characteristics on a per-channel level.

A repeatable channel element MAY therefore be defined as a child of a remoteContent element.

Each logical channel MAY have a local identifier (*chnid*), an indication of the media type of the data conveyed by the channel and an indication of the role the data plays in the scope of the full content, e.g. "voice over".

Each logical channel MAY be additionally described by the news content characteristics corresponding to the media conveyed in the channel.

4 Representing Concept Information - concept Component

4.1 Concept Component

Concepts fall in two broad categories: named entities and generic (or abstract) concepts. Generic concepts range from themes (e.g. soccer) to emotions (e.g. smiling, love); they have no specific property defined, beyond generic properties. Named entities are people, organisations, geographical areas and points of interest for which a specific set of properties is defined for the purpose of a refined definition and improved search and processing capabilities.

The concept element provides a set of properties shared by all concepts.

A concept can be identified in different schemes by different controlled values, this is why a concept identifier must be unambiguous, but cannot be unique: for example, a company is commonly identified by different ticker symbols. In the case of abstract topics, the strict sameness of two concepts may be subject to discussion, and therefore a notion of equivalence of concepts is preferred.

A concept MUST have a concept identifier, expressed as conceptId child elements.

The conceptId element MUST have a *qcode* attribute. It MAY have a *created* attribute and a *retired* attribute which limit the usage of the concept identifier in time.

A concept MAY have a type child element. The type of a concept reflects its nature, e.g. generic, person, organisation, geopolitical area, point of interest etc…

A concept MAY have a name and MAY be further defined in natural-language by a definition and note. These labels are repeatable and MAY be specified in multiple languages.

More detailed facets of a concept (e.g. that the concept "is" an artist, listed company, city, restaurant) MAY be expressed by a specific facet property. The facet property MUST have a *rel* attribute which specifies the exact facet which is described.

Different variants of a name are allowed. The *role* attribute the semantic of the property and takes values like "usual", "official", "married" (for a person) "acronym" (for an organisation), "synonym", "adjectival" (e.g. French for France). The *part* attribute identifies the part of the name conveyed by the property, and takes values like "given", "family" (for a person). Definitions and notes also support a role, which takes values like "history", "change" (for a description), "editorial", "scope" (for a note).

The descriptive elements name, definition and note and facet MAY have *validfrom* and *validto* attributes which limit the use of the property in time.

A concept MAY have a set of alternative identifiers, expressed as sameAs child elements.

The sameAs element MUST have a *qcode* attribute. It MAY have *created* and *retired* attributes, *validfrom* and *validto* attributes which limit the relationship in time.

4.2 Relationships Between Concepts

A concept MAY be associated with other concepts in the form of a taxonomy (i.e. a hierarchy of concepts) or thesaurus (i.e. a set of concepts associated via standard relationships). A concept MAY have an indication of most standard relationships as broader, narrower and related child elements.

The broader, narrower and related properties MUST have a *qcode* attribute which identifies a concept. They MAY have a *type* attribute which reflects the nature of the associated concept, and MAY have one or more names.

The broader, narrower and related properties MAY also have *validfrom* and *validto* attributes which limit the relationship in time, and a *rel* attribute which details the name given to the relationship. Broader MAY have a *rank* attribute which specifies the rank of the concept among the children of a given broader concept.

4.3 Details on Specific Entities

Details associated with specific entities MAY additionally be defined. All have been chosen for their potential usefulness in the news industry.

personDetails include a date of birth (born) and date of death (died) a repeatable indication of affiliation with an organisation and contact information (contactInfo).

organisationDetails include a date of foundation (founded) and date of dissolution (dissolved), a repeatable location and contact information (contactInfo).

The registered address of an organisation is indicated as part of its contact information; in such a case this address may not be used for making direct contact with this company.

geoAreaDetails include the geographic coordinates (position) of the place.

The position MUST have *latitude* and *longitude* attributes. It MAY have an indication of the altitude above the zero elevation reference level.

It MAY have an indication of coordinate reference system (*gpsdatum* atribute) expressed as a string. In the absence of this attribute, the WGS84 system is assumed.

POIDetails include the geographic coordinates (position) of the place, plus practical information like opening hours (openHours), capacity, access information, details on the location (e.g. room number, stair number etc.), and contact information (contactInfo).

4.3.1 Contact Information

Contact information (contactInfo) is repeatable in the definition of a person, an organisation and a Point of Interest, and each set of properties supports a *role* attribute which makes possible to group together all information belonging of the same nature.

Contact information include email addresses, instant messaging addresses (im), international phone numbers, international fax numbers, web addresses and postal addresses.

E-mail and instant messaging addresses, phone and fax number are all electronic addresses. These are qualified by a *role* attribute which specifies the nature of the address, e.g. home or work.

4.3.2 Postal Address

The definition of a Postal Address includes free-text lines (in the format expected by a recipient postal service), the indication of a locality (i.e. city, town, village etc…), a subdivision of a country (area), a country and a postal code (postalCode).

A postal address is structured as a set of properties likely edited and displayed as a form. The relative order of its properties is not universal, and if used for traditional postal mail, presentation algorithms are to be developed which depend on the source and recipient countries.

The city, country area and country may be indicated as a name or as a controlled value. The use of an ISO compliant country code is particularly recommended.

5 Managing Individual Concepts - conceptItem

An XML Schema file corresponding to the specifications for this item is available (see **The Full Set of Specification Documents** on page 3).

5.1 Description

A conceptItem aims to convey knowledge about a unique concept (a named entity such as an organisation or an abstract notion such as a news subject). Typically a conceptItem itself holds only rather short and structured information about the concept and about its relationships with other concepts.

Typical characteristics of a conceptItem are:

♦ It focuses on a single concept or entity.

♦ It will usually be updated infrequently but over a long period of time, when the information about the concept evolves.

♦ Its content is of long term interest.

♦ It may be referenced by other items.

Different conceptItems, managed by different providers, may contain structured information about the same concept.

5.2 Structure of a conceptItem

The model of a conceptItem is very similar to the model of a newsItem. Both share the same Indication of compliance with a standard and conformance level, Identification and versioning, Signature, Rights Information, Item Metadata, Item links. Please review the corresponding specification of a newsItem for more information.

Note that the globally unique guid of a conceptItem, which is used for management purposes, must not be confused with the unambiguous concept identifier (conceptId) defined by an authority and conveyed as part of the content of the Item.

5.3 Item Metadata

The IPTC provides a standardised scheme applicable to the itemClass property, identified by the URI: http://cv.iptc.org/newscodes/cinature/.

5.4 Concept related Metadata

The set of administrative metadata is common to all classes of Items. Please review **Representing News - newsItem** (page 20) for more information.

A conceptItem does not support descriptive metadata.

5.5 conceptItem Content

The content of a conceptItem is a concept component.

6 Managing Sets of Concepts - knowledgeItem

An XML Schema file corresponding to the specifications for this item is available (see **The Full Set of Specification Documents** on page 3).

6.1 Description

A knowledgeItem bundles a set of concept components which are managed and exchanged as a whole.

A knowledgeItem is used best where a provider wants to circulate a snapshot of a set of entries from one or more controlled vocabularies.

The concepts represented in a knowledgeItem can be of different types, and their identifiers may come from different schemes. A "scheme definition" is therefore a particular case of structure, where all concepts support a concept identifier from a same specific scheme.

Examples of knowledgeItems are the taxonomy of IPTC Subject NewsCodes or an authority list of people's descriptions maintained by a given provider.

Typical characteristics of a knowledgeItem are:

- ♦ It contains a meaningful set of concepts components.
- ♦ It will usually be updated infrequently but over a long period of time, for example when a controlled vocabulary evolves.
- ♦ Its content is of long term interest.

6.2 Structure of a knowledgeItem

The model of a knowledgeItem is very similar to the model of a newsItem. Both share the same Indication of compliance with a standard and conformance level, Identification and Versioning, Signature, Rights Information, Item Metadata, Item links. Please review **Representing News - newsItem** (page 20) for more information.

6.3 Item Metadata

The IPTC provides a standardised scheme applicable to the itemClass property, identified by the URI http://cv.iptc.org/newscodes/cinature/.

6.4 Knowledge Related Metadata

The set of administrative metadata is common to all classes of Items. Please review the corresponding specification of a newsItem for more information.

The set of descriptive metadata is listed below. All properties are optional, repeatable and may be inserted in any order.

Table 5. Knowledge Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Subject	**subject**	(0..unbounded)	193
Description	**description**	(0..unbounded)	98

Each provider may add a set of metadata properties which have to be defined in a non-IPTC namespace. See also **XML Namespaces** (page 44) and **Extension Points in XML** (page 45).

6.5 knowledgeItem Content

A conceptSet wrapper element contains a set of concept components. Their order of appearance in conceptSet is not relevant.

Note: All concept definitions share the same catalog of schemes, declared at the top of the knowledgeItem.

7 Packaging Items - packageItem

An XML Schema file corresponding to the specifications for this item is available (see **The Full Set of Specification Documents** on page 3).

A packageItem facilitates the packaging of all kinds of Items, from really simple constructs to the highly hierarchical structures created by some news providers.

Examples of packageItems are a collection of pictures, a "top ten" list of newsItems, an unordered set of newsItems relative to the same event, the representation of a newspaper section or page.

Typical characteristics of a Package Item are:

♦ It provides some structure to news related information, and is expressed via a hierarchy of items.

♦ The Items found in a packageItem stay independent from the package: they can be managed individually, and the package keeps only references to them.

♦ Its content is of medium term interest.

7.1 Structure of a packageItem

The model of a packageItem is very similar to the model of a newsItem. Both share the same Indication of compliance with a standard and conformance level, Identification and versioning, signature, rights information, Item metadata, Item links. Please review the corresponding specification of a newsItem for more information.

7.2 Item Metadata

The IPTC provides standardised schemes applicable to the itemClass property of a packageItem, identified by the URI http://cv.iptc.org/newscodes/ninature/ and http://cv.iptc.org/newscodes/cinature/.

7.3 Package Related Metadata

The set of administrative and descriptive metadata is common between packageItems and newsItems. Please review **Representing News - newsItem** (page 20) for more information.

7.4 packageItem Content

A groupSet represents a tree of Items. All Items of a package are included by reference, as physical inclusion would break the capability to manage inner Items independently of the outer Package Item.

The groupSet is optional. This allows for a lightweight and progressive representation of information.

There MUST be at least one group element in the groupSet. If there is more than one group in the groupSet element, a specific group acts as the root of the tree. In this case the value of *root* attribute of the groupeSet element MUST be the local reference to the group acting as a root.

A group component contains references to other group components (groupRef with its *idref* attribute) and references to Items or Web resource (itemRef with its *guidref* and *href* attributes), in any order.

Each group MAY have a *role* attribute which indicates the part this group plays within its container.

The order of the sub-groups and references to Items MAY be significant; a *mode* attribute indicates whether the elements in the group are complementary and unordered, complementary and ordered or a set of alternative elements. In the absence of a *mode* attribute the group is treated as complementary and unordered.

The itemRef element MAY contain metadata extracted from the target Item or Web resource. All metadata SHOULD be included directly as children of the itemRef element, without any wrapper element (e.g. neither itemMeta nor contentMeta). The recipient MUST NOT consider that such hints constitute a complete representation of the Item.

The itemRef element MAY have a *rank* attribute which represents the rank of the Item among other Items in each group.

The itemRef element MAY also have time validity attributes (*validfrom* and *validto*) which express the date and time between which the reference is active.

Sample:

```
<groupSet root="g1">
<group id="g1" mode="mode:seq" role="group:main">
<groupRef idref="g2"/>
<itemRef guidref="urn:newsml:iptc.org:20070530:tutorial-text-xhtml"/>
</group>
<group id="g2" role="group:gallery">
<itemRef guidref="urn:newsml:iptc.org:20070530:tutorial-iptc-logo"/>
<itemRef guidref="urn:newsml:iptc.org:20070530:tutorial-video"/>
</group>
</groupSet>
```

8 Dealing with Controlled Values

8.1 {scheme, code} Pair, Scheme URI and Concept URI

Many properties usually have their value taken from a well defined scheme, i.e. a controlled vocabulary (i.e. a classification system, authority list, taxonomy, thesaurus etc ...).

These values are represented by a formal combination - a {scheme, code} pair - primarily intended to be consumed by processing software. A scheme is logically a closed set of related concepts, and a {scheme, code} pair unambiguously identifies a single concept.

A scheme is in practice a list of codes managed by a specific authority (which we shall refer to as the Scheme Authority), which may be the IPTC or any other well known standardisation body, or may be an individual news provider. A {scheme, code} pair therefore fully identifies a term from a controlled vocabulary (i.e. a scheme). A code MUST be persistent over time in order to avoid ambiguities when processing archives documents.

A scheme is fully and unambiguously identified by a scheme URI. The concept represented by a code is fully and unambiguously identified by a concept URI. The concept URI is obtained by appending the code to the scheme URI. **Qualified Code (QCode)** (page 38) shows how a more compact form of a concept identifier is used in the news workflow.

As an example, an IPTC scheme for news categories might be identified by the URI "http://cv.iptc.org/newscodes/theme/". If the code "15000000" represents the concept of "Sport", then the concept URI for "Sport" would be "http://cv.iptc.org/newscodes/theme/15000000".

It is not mandatory that the Scheme Authority maintains the complete list of codes making up a given scheme in any particular form, e.g. as an XML file. It is sufficient that an unambiguous identifier is defined for each scheme a provider uses, and that this identifier is known to the customers of the news feed this provider offers.

Common needs are:

♦ To access human readable information about a scheme.

♦ To retrieve all terms of a scheme (e.g. to display a list of choice).

♦ To access human readable information about a qualified code.

♦ To check that a qualified code belongs to a scheme.

♦ To retrieve the definition of the concept identified by a qualified code in a given scheme.

Therefore the scheme URI SHOULD resolve to a resource (or resources) containing information about the scheme in both human-readable and machine-readable forms. Meeting this requirement is mandatory for schemes which are to be compliant with the Semantic Web.

The concept URI SHOULD also resolve to a resource (or resources) containing information about the concept in both human-readable and machine-readable forms. Meeting this requirement is mandatory for concept URIs which are to be compliant with the Semantic Web.

If content negotiation is implemented using HTTP, then the HTTP Accept header should be used to request information in the required format and the HTTP Accept-Language header should be used to request information in the required human language.

When designing a scheme URI, the following points should be taken into consideration:

♦ Each scheme URI must end with a suitable terminating character, e.g. "/" or "#". Each of these has various advantages and disadvantages, which are discussed extensively in documents available on the Web.

♦ One point worth mentioning here is that not all strings which can be used to construct a legal URI are automatically legal in the context of HTML. For example, "http://cv.iptc.org/newscodes/theme.html#15000000" is not a legal HTML URI, as an HTML fragment name cannot start with a digit.

8.2 Qualified Code (QCode)

In order to manipulate controlled values in an efficient manner, a compact representation of a concept identifier is needed, a syntax which allows the use of a {scheme, code} pair as the value of an XML attribute.

For this purpose a short string called scheme alias (aka prefix) is defined by a provider as a substitute for a scheme URI in the scope of a single Item, and a compact notation of a scheme-code pair is defined, called qualified code or QCode.

The datatype for a compact notation of a scheme-code pair is called qualified code or more simply QCode. QCodes are the mandatory way to express controlled values in properties like itemClass or pub-Status.

QCodes are notated by the following syntax: a scheme alias acting as a first part, followed by a colon (:) character, followed by a code from the scheme. They are case sensitive.

The value space of the QCode Type datatype is a set of {scheme, code} pairs which identify concepts.

Note that:

♦ This is similar to the value space of the QName datatype, i.e. a set of {namespace, local part} pairs which identify element or attribute names.

Note: QNames cannot be used for this purpose, because the local part of QNames cannot be a numeric, but the News industry and the Financial industry are full of taxonomies making use of numeric codes. They aren't alone in this aspect (consider ISBN and ISSN).

♦ QCodes allow any sequence of legal URI characters in the code part, including, for example, digits only, dashes, slashes, etc.

♦ QCodes MUST have a non-empty scheme alias.

QCodes can be viewed to a certain extent as short, lexical representations of URIs. Be careful: the mapping from a qualified code to a URI is not bijective: a URI cannot be mapped back to a qualified code, because the separator of the tuple is not explicitly defined in the URI.

In order to resolve a qualified code, a processor MUST loop through the scheme elements defined in the scope of the Item. If the QCode scheme alias is found as value of the *alias* attribute of a scheme element, the scheme URI is the associated *uri* attribute and the controlled value is the resulting {scheme URI, code} pair. If no corresponding scheme alias is found, the processor SHOULD raise an error and consider that the property has no value.

8.2.1 Processing QCodes

For processing QCodes properly a processor has to comply to these three rules:

♦ Syntax Space for Aliases: all characters except colon (#x3A) and white space (#x20 | #x9 | #xD | #xA).

♦ Syntax Space for Codes: all characters except white space (#x20 | #x9 | #xD | #xA).

♦ Processing Rule: To split a QCode term into its alias and code part the parser has to identify the first colon searching from left to right. If it encounters one the string to the left is the alias, the string to the right is the code. If it encounters none the QCode is invalid.

8.3 Processing Catalogs

8.3.1 Structure of a Catalog

A catalog MUST have one or more scheme elements.

A scheme element MUST have a scheme *alias* attribute and a corresponding scheme *uri* attribute.

Each Item defines its own set of scheme definitions, and there is no interaction between scheme definitions in different Items. Scheme alias declarations are local to the Item in which they appear and cannot be overridden in a given Item.

8.3.2 Processing Remote Catalogs

By activating the hyperlink of a remoteCatalog, a plain catalog structure is returned, and MUST be processed as if were locally defined.

8.3.3 Caching a Catalog

It is recommended for a processor to cache a remote catalog indefinitely, so that provider's servers are not overcharged with file requests.

When a processor opens an Item, it MUST check the URL(s) of the catalog(s) found in its header. If a catalog has not been previously cached, the processor MUST fetch it, check it, and SHOULD store its content in a cache.

Different remote catalogs MAY define different mappings for a given scheme alias. An entry in a remote catalog cache is therefore a triple {remote catalog URL, scheme alias, scheme URI}.

8.3.4 Checking a Catalog

It is OK for one scheme URI to have two aliases. It is an error if one alias is mapped to two different URIs in the scope of a single Item (an issue called alias collision). Note that this error may arise within a catalog, as well as across a set of catalogs (local or remote) declared in a given Item.

Before processing an Item, a processor MUST check its catalogs. If an alias collision is found, the processor MUST reject the Item as it can lead to misinterpretation of the information.

Note: If an aggregator finds an alias collision (i.e. the same alias associated with two URIs) while creating a packageItem which aggregates content from various providers, the aggregator MUST change one or both of the aliases before publishing the packageItem. This can be done by creating and publishing one or more non-clashing external catalogs (which replace the original external catalogs) and/or by replacing one or more external catalogs with non-clashing in-line scheme declarations.

8.4 Processing Schemes

8.4.1 Evolution of Scheme URIs

Schemes evolve: terms are added, names are changed, terms are retired. An authority will release a new version after each update. A provider may not want to adopt the latest version of a scheme. The scheme URI MUST be stable as long as the evolution does not break backward compatibility rules.

8.4.2 Retrieving All Terms of a Scheme

Here we are interested in schemes defined as an explicit list of terms. Schemes defined via an algorithm are out of scope of this section. A scheme definition is defined as the finite set of terms composing a scheme. A scheme definition MAY be a subset of an original scheme, e.g. maintained by an external authority.

Note: An authority is not necessarily able to make scheme definitions available for operational use, and a provider may use only a subset of the scheme defined by an authority.

A provider SHOULD make a scheme definition available for its users for operational use as the content of a knowledgeItem, where each term is represented as a concept component, i.e. a concept identifier, a list of names in one or more languages, plus additional properties of the concept (all but the identifier being optional).

An authority MAY provide different variants of a scheme definition, e.g. a list of codes, a list of codes plus a name in a specific language, a list of codes plus names in all available languages.

For each variant of a scheme definition, the URL of the corresponding knowledgeItem SHOULD be available using e.g. content negotiation.

Selection from among the renditions MAY be performed automatically (if the processor is capable of doing so) or manually by the user selecting from a hypertext menu.

8.5 Qualified and Typed Properties

Qualified properties – of datatype Qualified Property Type – only support controlled values in the format of QCodes.

A large subset of these properties supports concepts of different types as a value. Therefore typed qualified properties – of datatype Typed Qualified Property Type – additionally provide a concept type relative to the value of the property.

For example, the type of the concept assigned as subject of a news story may be a theme (e.g. sport or football), a person, an organisation, a geographical area, a point of interest, an event, a business sector, a currency etc. The concept type of a creator, contributor and infoSource of an Item may be a person or an organisation

Qualified properties MAY be complemented by one or more names associated with the underlying concept. Names can be expressed in different languages or variants.

8.6 Flexible Properties

It is not always possible or sensible to use a QCode as metadata value. As an example, few news organisations maintain a formal listing of their editors, and therefore using a controlled value for the creator property is not always possible.

In order to fulfil this need, a large number of properties allow for literal values – i.e. uncontrolled free-text values – to be applied instead of controlled values.

Therefore flexible properties – of datatype Flexible Property Type or a derived datatype – support both controlled (qcode) and uncontrolled (literal) values.

QCodes and literals are mutually exclusive for one property, but either one or the other MUST be indicated.

The value of a flexible property identifies a given concept with a specific type. It is useful to express e.g. that the provider of a news item is a person or an organisation. The *type* of the concept MAY be indicated as an attribute of the flexible property.

One or more additional name properties MAY be provided in different languages and variants for display purpose.

Such metadata MAY are usually given when the property value is a QCode, but MAY also be provided when the property value is a literal. If the value is a literal and no additional name is given, the recipient MAY use the literal value for direct display.

Flexible properties MAY also be complemented by other information about the concept, like properties from Concept Relationships Group and Concept Definition Group.

Flexible properties which value specifically identifies a person, an organisation or any other entity for which detailed properties are defined in this specification, MAY contain detailed information about this entity, e.g. a date of birth for a person of a location for an organisation.

Such information constitutes "hints" about the concept, which may be useful for display or indexing, but which should not be used to convey knowledge stored as-is in a knowledge repository. A specific mechanism, based on conceptItems and knowledgeItems, is set-up in the News Architecture for managing knowledge.

8.7 Composite Concepts

Several flexible properties support composite concepts. Composite concepts, a.k.a. pre-coordinated terms, are "glued" together to represent a concept made of atomic parts.

Therefore flexible concept properties – of datatype Flexible 1 Concept Property Type – have a bag child element which is used to express a new concept, composed from multiple existing concepts. The description of each existing concept is placed in a bit child element of the bag wrapper.

Examples of possible composite concepts are:

- ♦ John Doe Smiling {John Doe + Smiling }
- ♦ Women's 100m Swimming Final {Women + Swimming + 100m + Final}
- ♦ Positive pre-announcement by Citigroup {Citigroup + Pre-announcement + Positive}
- ♦ Microsoft's share price has moved up {Microsoft + Share price + Up}
- ♦ The Clintons {Bill Clinton + Hillary Clinton}

8.8 Editing Attributes

In a professional and collaborative news workflow, it makes sense to identify all elements defined by the model in order to later act on them individually. Also, metadata is not always entered by one person at one time, but may be entered by different people, organisations or systems at different time. Therefore it may be needed to keep track of who is assigned the editing of specific properties, and when and by whom a property has been given a value.

For this purpose, all metadata properties share an Editing Attributes Group, i.e. an optional local identifier (id) and the optional indication of the creator and the date (and, optionally, the time) when the property was last modified.

9 Dealing with Labels and Blocks

9.1 Introduction

Labels expose aspects of news as natural language strings. They are assertions expressed as natural language strings intended to be consumed by human beings. They are typically displayed alongside the content of an Item or in place of Items in a list, providing a means of selection among them.

Blocks are simply labels with an additional line break. They are primarily used for notes, comments or instructions created by a news provider for use by recipient editorial teams.

Labels and blocks MAY have a *role* attribute, which refines the semantics of the property.

Labels and blocks MAY have a *media* attribute. When present, the value MUST conform to the Cascading Style Sheets specification [CSS]. Several media types can be given as space separated values.

All labels and blocks support rich text, i.e. text interspersed with some specific markup, identical to XHTML1.1 markup: the anchor (a) for the inclusion of hyperlinks, the span as a generic mechanism for adding information to text, simple ruby markup used in Japanese publications and inline for semantic inline markup.

The inline property identifies a concept present in a label or block either by a qualified code or a literal value, plus an optional type. Additional information about this concept can be represented using an assert property.

9.2 Internationalization Attributes

In an international news workflow, fine grained control of language information in the hierarchy of nodes constituting an Item is needed.

For this purpose, all labels – and all ancestors of such an element – share an International Attributes Group, i.e. an optional language tag (xml:lang) and indication of the directionality of textual content (dir).

10 Exchanging Items - newsMessage

A newsMessage facilitates the exchange of all kinds of items by any kind of digital transmission, especially in a broadcast or multicast network.

The content of a newsMessage is an itemSet component.

An itemSet contains a set of newsItems, packageItems, conceptItems and knowledgeItems. The model assigns no significance to the order of Items within the News Message.

The use of a News Message is totally optional in a news workflow. Alternatively, Items may be exchanged using SOAP, WebDAV, ICE, the Atom Publication Protocol (using Atom feeds, and items as payload of an Atom entry) or any other possible syndication protocol.

It may be useful for a recipient to store the information conveyed by a message, but this is not mandatory. Usually the messaging information will be maintained separately from the information relative to the contained items.

10.1 Message Information

A newsMessage MUST have a date of transmission (sent). The date of transmission MAY not be updated in case of retransmission of the message.

A newsMessage MAY have a sender child element, which may be an organisation or a person. The structure of this string is not specified by the IPTC. Best practice is to identify a sender by its domain name

It MAY have a transmission identifier (transmitId) and a priority of transmission. No two newsMessages sent by the same sender on the same date can have the same identifier. In case of retransmission it is not required to update this identifier. The structure of this string is not specified by the IPTC.

It MAY indicate the point of origin of the message, using a provider defined syntax.

It MAY have one or more timestamp(s) associated with the message. The exact meaning of this timestamp may be refined by a *role* attribute.

It MAY have one or more destination(s) using a provider defined syntax, and the indication of one or more channel(s) of transmission.

Each particular provider is equally able to add to this set information of its own, by mutual agreement with the recipients of the Item.

10.2 About Using Schemes in a newsMessage

It is important to note that a newsMessage does not define any catalog that would be common to the Items it contains. There is no interaction between the scheme declarations present in different Items exchanged in a newsMessage.

11 Specification Reference

This section provides all specifications for this G2-Standards, the different specifications tables are cross referenced from other parts of this document.

11.1 Introduction to the Common Components

News exchange formats share many metadata properties as they are about the same data: something newsworthy to be exchanged. For that reason the family of IPTC G2-Standards share a large set of properties which are common to all family members and this common data model and set of specifications is called the IPTC News Architecture for G2-Standards (NAR).

This Specification Reference section provides a mix of specifications coming from the NAR and additionally from this G2-Standard.

The components specified in this Specification Reference can be split into these 3 groups:

1. Fine grained components, called a datatype. A datatype has no specific business meaning or semantics of its own and only takes on business meaning when used as the data type of a property. Datatypes fall in two groups:

  ● Simple data types are primitive data types, as found in software languages or XML schema definitions (e.g.. integer, string). Some restriction may be imposed, such as Int100Type where an integer has been restricted to a value range of 0 to 100.

  ● Complex data types are simple data types extended to add further information in order to correctly represent the value. Such ancillary information takes the form of attributes. For example a Label-Type supports mixed content and is extended with language and role attributes.

For a G2-Standard the names of datatypes end with a "Type" suffix (e.g. QCodeType).

2. Medium grained components, called basic component or property. A property represents a single piece of business information and uses an existing data type or defines it own local datatype to provide its content model. It is capable of being used independently or as part of a group. Like a complex data type, a basic component can be qualified by ancillary data if required to complete its meaning. For example, a slugline element of data type string supports an additional separator attribute.

3. Coarse grained components, called aggregate component. It is a collection of properties that together is more than the sum of its constituent parts. The properties composing the whole can be properties or aggregate components. An aggregate component may be designed so it supports an extension point where news providers can extend its usage. For example, a descriptive component is defined as a group of properties like title and subject, and a person component is defined as a group of properties like name and date of birth.

11.2 General Specifications

11.2.1 XML Namespaces

Table 6. XML Namespaces

Namespace URI	Recommended Alias	Usage Note
http://iptc.org/std/nar/2006-10-01/	nar	For all common components of the IPTC G2-Standards.

11.2.2 MIME Types

Namespace URI	Usage Note
application/vnd.iptc.g2.newsitem+xml	For all kinds of G2 News Items.
application/vnd.iptc.g2.conceptitem+xml	For all kinds of G2 Concept Items.
application/vnd.iptc.g2.packageitem+xml	For all kinds of G2 Package Items.
application/vnd.iptc.g2.knowledgeitem+xml	For all kinds of G2 Knowledge Items.

11.2.3 Extension Points in XML

For attributes: each element of a G2-Standard allows to add provider specific attributes from any other XML namespace than the News Architecture for G2 namespace (see **XML Namespaces** on page 44).

For elements: Some elements which have child elements allow to add provider specific elements from any namespace other than the News Architecture for G2 namespace (see **XML Namespaces** on page 44). A few elements allow adding any element from any XML namespace - including the News Architecture for G2 namespace - but this is a special case only.

11.3 Implementation Design Rules

These design rules were applied while developing the G2-Standards. Some apply to all kinds of technical implementations, other only to one specific implementation. Further some rules are only applicable at one of the Conformance Levels CCL or PCL.

♦ Each element supports editing attributes (PCL).

♦ Each element has an extension point at the attribute level (XML implementation only).

♦ Each element containing an international string supports i18 attributes (CCL).

♦ Each ancestor of an element containing an international string supports i18 attributes (PCL).

♦ Children of wrapper elements: mandatory children come first, they are in a specific order, optional (and in most case multiple) elements follow, they can be inserted in an arbitrary order (XML implementation only).

♦ Each wrapper element has an extension point as its last child element (XML implementation only).

11.4 Processing Model Terminology

For many components of the G2-Standars this specification provides also a processing model. Find below how these processing instructions should be read.

♦ A Processing Model provides rules for the proper processing of metadata properties and their values. Each rule may be divided into steps.

♦ Each rule gets an integer number assigned, steps for this rules are indicated as decimals to this number. Example: rule 12, step 3 = 12.3

♦ Many rules can be considered like a function in programming, hence as a sequence of processing steps in the scope of a block. These terms will be used for defining the rules and are based on this basic layout:

    ● "quit" = the processing of this function stops at this step and quits the current context to the calling context.

    ● "quit and return ..." = see "quit", plus: a value of "..." is returned to the calling context.

    ● "if ... :" = a condition is expressed and right to the colon the processing that results from meeting this condition.

    ● If the condition is NOT met the default processing is "proceed to the next step of this processing rule". A specific processing for this case is preceded by the term "otherwise".

11.5 Component Structure Format

Table 7 describes the component (element and datatype) specifications of the G2 data model. This table is divided into two sections:

- ♦ The upper section contains the specification of generic properties of the component.
- ♦ The lower section(s) contain the specification of the component based on the W3C XML Schema 1.0 (XMLSCHEMA-1.0) specifications. This section may contain different specifications at the Core Conformance Level (CCL) and the Power Conformance Level (PCL) of the G2 data model.

Descriptions of the individual specifications can be found in **Table 7**.

Table 7. Component Structure Format

(XML) Data Model	Defines a high -level data model for this component. The value is one out of: simple datatype/complex datatype/element/attribute group.
Namespace (Prefix)	Namespace for the name of this component. Is either *nar* for the generic G2-Standards namespace or a prefix for any other namespace. Which prefix is assigned to which namespace is defined by a heading section of a G2-Standard specification document.
Name	The technical reference of the component (must align with the name in the XML Schema!).
Title	The natural-language label of the component.
Definition	A concise definition of the semantics of the component.
User Note(s)	Any notes addressing the (end-)user of the component with a focus on its proper use.
Implementation Note(s)	Any notes addressing the implementer of the component into any technical system.
XML Schema Spec	At: Both CCL and PCL / CCL / PCL; indicates at which conformance levels this XML Schema specification applies.
Datatype	The XML Schema datatype or any datatype defined by a G2-Standard.
Internally Ctrl Values	A definition of one or more values if they are controlled by the XML Schema, e.g. an enumeration or regular expression.
Externally Ctrl Values	A definition of any controlled vocabulary with values to be (exclusively) used with this component.
Attribute(s)	One or more XML attributes defined for this component if it is either a complex datatype, an element, or an attribute group.
Child Element(s)	One or more child elements defined for this component if it is either a complex datatype or an element.
XML Schema Note(s)	Any notes regarding the implementation of this component into the XML Schema of this G2-Standard.
Example(s)	One or more XML snippets showing use-cases for this component.

11.6 Element Definitions

11.6.1 Access

Table 8. Access

(XML) Data Model	Element
Namespace (prefix)	nar
Name	access
Title	Access
Definition	Ways to access the place, including directions.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**BlockType** (page 208)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.2 Access Status

Table 9. Access Status

(XML) Data Model	Element
Namespace (prefix)	nar
Name	accessStatus
Title	Access Status
Definition	An indication of the accessibility of the event.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QCodePropType** (page 233)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.3 Accountable Person

Table 10. Accountable Person

(XML) Data Model	Element
Namespace (prefix)	nar
Name	accountable
Title	Accountable Person
Definition	An individual accountable for the content in legal terms.
User Note(s)	This property answers to a legal issue. In some countries (e.g. Germany, Sweden) it is needed to designate a person accountable for any legal issue related to the published content. The full translation from the German term is: accountable person in terms of the press law - (For reference in German: Verantwortlich im Sinne des Presserechts -acronym = ViSdP), in Swedish it is called "Ansvarig utgivare". In practice today, a news provider may send out a message each day which indicates the "accountable person". This may work for traditional feed services, but fails with profiled services (content selections) which filter such messages. The solution is to include this information in the Items themselves.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**FlexPersonPropType** (page 224)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.4 Address Line

Table 11. Address Line

(XML) Data Model	Element
Namespace (prefix)	nar
Name	line
Title	Address Line
Definition	A line of address information, in the format expected by a recipient postal service. City, country area, country and postal code are expressed separately.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**IntlStringType** (page 227)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.5 Affiliation

Table 12. Affiliation

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	affiliation		
Title	Affiliation		
Definition	An affiliation of the person with an organisation.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	**FlexOrganisationPropType** (page 221)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 244)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.6.6 Alternative Identifier

Table 13. Alternative Identifier

(XML) Data Model	Element
Namespace (prefix)	nar
Name	altId
Title	Alternative Identifier
Definition	Alternative identifier allocated to the content.
User Note(s)	If there is more than one alternative identifier, they SHOULD be qualified using the type qualifier to distinguish between different identification schemes.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**IntlStringType** (page 227).
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ type (0..1); **QCodeType** (page 234); A qualifier which indicates the context within which the alternative identifier has been allocated.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.7 Alternative Representation

Table 14. Alternative Representation

(XML) Data Model	Element
Namespace (prefix)	nar
Name	altRep
Title	Alternative Representation
Definition	An IRI which, upon dereferencing provides an alternative representation of the Item.
User Note(s)	This property is particularly useful if the Item is available in different formats (for example NewsML 1, IIM or NITF) or with different levels of details (for instance with different granularity of metadata).
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**IRIType** (page 228)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ representation (0..1); **QCodeType** (page 234); A qualifier which specifies the way the targetItem is represented at this location. ▪ **editAttributes** (page 243)
Child Element(s)	
XML Schema Note(s)	
Example(s)	

Attribute table within Attribute(s) row:

Name	Datatype
id (0..1)	XML Schema ID
creator (0..1)	QCodeType
modified (0..1)	DateOptTimeType

11.6.8 Anchor

Table 15. Anchor

(XML) Data Model	Element
Namespace (prefix)	nar
Name	a
Title	Anchor
Definition	An anchor for inline linking like in HTML.
User Note(s)	
Implementation Note(s)	This element is modelled after its XHMTL 1.0 counterpart.
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ class (0..1); XML Schema String; An equivalent of the HTML class attribute.

		Name	Datatype
	▪ href (0..1); **IRIType** (page 228); An equivalent of the HTML href attribute.		
	▪ hreflang (0..1); XML Schema NMTOKEN; An equivalent of the HTML hreflang attribute.		
	▪ rel (0..1); XML Schema string; An equivalent of the HTML rel attribute.		
	▪ rev (0..1); XML Schema string; An equivalent of the HTML rev attribute.		
	▪ **editAttributes** (page 243)	id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 243)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	▪ **span** (page 191) (0..unbounded)
	▪ **ruby** (page 182) (0..unbounded)
	▪ **inline** (page 124) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	Implementation of the attributes aligns with the XHTML 1.0 specs.
Example(s)	

11.6.9 Assertion

Table 16. Assertion

(XML) Data Model	Element
Namespace (prefix)	nar
Name	assert
Title	Assertion
Definition	An assertion about a concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ qcode (1); QCodeType. A concept identifier.
Child Element(s)	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

The Attribute(s) cell also contains the following tables:

editAttributes (page 243)

Name	Datatype
id (0..1)	XML Schema ID
creator (0..1)	QCodeType
modified (0..1)	DateOptTimeType

i18nAttributes (page 243)

Name	Datatype
xml:lang (0..1)	XML Schema language
dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

11.6.10 Audience

Table 17. Audience

(XML) Data Model	Element
Namespace (prefix)	nar
Name	audience
Title	Audience
Definition	An intended audience for the content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**AudienceType** (page 207)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.11 Bag

Table 18. Bag

(XML) Data Model	Element
Namespace (prefix)	nar
Name	bag
Title	Bag
Definition	A group of existing concepts which express a new concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	

Attribute(s)		Name	Datatype
	editAttributes (page 243)	id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	i18nAttributes (page 243)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	▪ **bit** (page 58) (1..unbounded)
XML Schema Note(s)	
Example(s)	

11.6.12 Bag Item

Table 19. Bag Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	bit
Title	Bag Item
Definition	An individual concept, part of a composite concept expressed by a bag.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**QCodePropType** (page 233)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.13 Broader

Table 20. Broader

(XML) Data Model	Element
Namespace (prefix)	nar
Name	broader
Title	Broader
Definition	An identifier of a more generic concept.
User Note(s)	*rank* (available at the PCL only) is suitable for use in a Knowledge Item representing a scheme. It is used when it is important that the Child Elements of a particular term are displayed in a user interface in a predefined order. For example, the major currencies could be given a rank of "1", while all other currencies could be given a rank of "2". Terms of the same rank are ordered alphabetically by name if this is available. If the name is not available, the terms are ordered by code value. Terms without a rank are treated as if they all have the same rank, which is higher than the rank of all other terms. The same concept may have different ranks in different concept trees. A lower rank results in a placement earlier in a display.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**RelatedConceptType** (page 237)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ rank (0..1); XML Schema nonNegativeInteger; Specifies the rank of the concept among the children of a given broader concept.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.14 By

Table 21. By

(XML) Data Model	Element
Namespace (prefix)	nar
Name	by
Title	By
Definition	A natural-language statement about the creator (author, photographer etc.) of the content.
User Note(s)	The *by* label provides a natural-language statement of the author/creator information (commonly called the byline); it may include a byline title, i.e. the author's job title. Examples of bylines are RUPAK DE CHOWDHURI (a person), isotype.com (a provider) or STR (a stringer). It is up to the provider to decide if the label starts with a word like "By".
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Label1Type** (page 229)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.15 Capacity

Table 22. Capacity

(XML) Data Model	Element
Namespace (prefix)	nar
Name	capacity
Title	Capacity
Definition	Total capacity of the place in natural language.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Label1Type** (page 229)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.16 Catalog

Table 23. Catalog

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	catalog		
Title	Catalog		
Definition	A local or remote catalog.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)	▪ **title** (page 138) (0..unbounded)		
	▪ **scheme** (page 187) (1..unbounded)		
XML Schema Note(s)			
Example(s)			

11.6.17 Channel of Remote Content

Table 24. Channel of Remote Content

(XML) Data Model	Element
Namespace (prefix)	nar
Name	channel
Title	Channel of Remote Content
Definition	Information about a specific content channel.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	

Table 24. Channel of Remote Content (Continued)

	▪ chnlid (0..1); XML Schema positiveInteger; A logical identifier of the channel		
	▪ type (0..1); **QCodeType** (page 234); The media type of the data conveyed by the channel.		
	▪ role (0..1); **QCodeType** (page 234); The role the data of this channel plays in the scope of the full content.		
		Name	**Datatype**
		wordcount (0..1)	XML Schema nonNegativeInteger
		width (0..1)	XML Schema nonNegativeInteger
		height (0..1)	XML Schema nonNegativeInteger
		orientation (0..1)	XML Schema nonNegativeInteger
		colourspace (0..1)	QCodeType
		resolution (0..1)	XML Schema positiveInteger
Attribute(s)		duration (0..1)	XML Schema nonNegativeInteger
	▪ **newsContentCharacteristics** (page 245)	audiocodec (0..1)	XML Schema nomalizedString
		audiobitrate (0..1)	XML Schema positiveInteger
		audiovbr (0..1)	enumeration: yes/no
		audiosamplesize (0..1)	XML Schema positiveInteger
		audiosamplerate (0..1)	XML Schema positiveInteger
		audiochannels (0..1)	QCodeType
		videocodec (0..1)	XML Schema nomalizedString
		videoavgbitrate (0..1)	XML Schema positiveInteger
		videovbr (0..1)	enumeration: yes/no
		videoframerate (0..1)	XML Schema positiveInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.6.18 Channel for News Message

Table 25. Channel for News Message

(XML) Data Model	Element
Namespace (prefix)	nar
Name	channel
Title	Channel for News Message
Definition	A transmission channel used by the message.
User Note(s)	A channel identifier is used to provide recipients with information for selecting, routing, or handling otherwise the content of the message. The channels represent streams in a multiplex: a message may be sent on different channels – e.g. one for text, one for pictures – and each reception point will be able to filter on channel values. The structure of this string is not specified by the IPTC.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.19 Concept

Table 26. Concept Definition

(XML) Data Model	Element
Namespace (prefix)	nar
Name	concept
Title	Concept
Definition	A set of properties defining a concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **conceptId** (page 68) (1)
	▪ **type** (page 197) (0..1)
	▪ **name** (page 70) (1..unbounded)
	▪ **definition** (page 67) (0..unbounded)
	▪ **note** (page 153) (0..unbounded)
	▪ **facet** (page 112) (0..unbounded)

▪ **Concept Relationships Group** (page 202) (0..1)	Element Name	Page
	broader (0..unbounded)	59
	narrower (0..unbounded)	149
	related (0..unbounded)	175
	sameAs (0..unbounded)	186

▪ **Entity Details Group** (page 203) (0..1)	Element Name	Page
	geoAreaDetails (1)	118
	organisationDetails (1)	157
	personDetails (1)	163
	POIDetails (1)	165

	▪ **eventDetails** (page 106) (0..1)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.20 Concept Definition

Table 27. Concept Definition

(XML) Data Model	Element
Namespace (prefix)	nar
Name	definition
Title	Concept Definition
Definition	A natural-language definition of the semantics of the concept. This definition is normative only for the scope of the use of this concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**BlockType** (page 208)
Internally Ctrl Values	
Externally Ctrl Values	

		Name	DataType
Attribute(s)	▪ **timeValidityAttributes** (page 244)		
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType

Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.21 Concept Identifier

Table 28. Concept Identifier

(XML) Data Model	Element
Namespace (prefix)	nar
Name	conceptId
Title	Concept Identifier
Definition	The preferred unambiguous identifier for the concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**ConceptIdType** (page 209)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.22 Concept Item

Table 29. Concept Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	conceptItem
Title	Concept Item
Definition	An Item containing information about a concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**AnyItemType** (page 205)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta** (page 74) (0..1) ▪ **assert** (page 55) (0..unbounded) ▪ **inlineRef** (page 128) (?) ▪ **concept** (page 66) (0..1)
XML Schema Note(s)	
Example(s)	

11.6.23 Concept Name

Table 30. Concept Name

(XML) Data Model	Element
Namespace (prefix)	nar
Name	name
Title	Concept Name
Definition	A natural-language name for the concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**IntlStringType** (page 227).
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the *part* attribute: http://cv.iptc.org/newscodes/namepart/
Attribute(s)	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the name. ▪ part (0..1); **QCodeType** (page 234); Specifies which part of a full name this property provides. <table><tr><td rowspan="3">▪ **timeValidityAttributes** (page 244)</td><td>**Name**</td><td>**DataType**</td></tr><tr><td>validfrom (0..1)</td><td>DateOptTimeType</td></tr><tr><td>validto (0..1)</td><td>DateOptTimeType</td></tr></table>
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.24 Concept Set

Table 31. Concept Set

(XML) Data Model	Element
Namespace (prefix)	nar
Name	conceptSet
Title	Concept Set
Definition	An unordered set of concepts.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **concept** (page 66) (0..unbounded)
XML Schema Note(s)	
Example(s)	

11.6.25 Confirmation

Table 32. Confirmation

(XML) Data Model	Element
Namespace (prefix)	nar
Name	confirmation
Title	Confirmation
Definition	Flag to indicate if start and/or end date and times are confirmed.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QCodePropType** (page 233)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes: http://cv.iptc.org/newscodes/eventdateconfirm/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.26 Contact Information

Table 33. Contact Information

(XML) Data Model	Element
Namespace (prefix)	nar
Name	contactInfo
Title	Contact Information
Definition	Information to get in contact with the entity expressed by the wrapping property.
User Note(s)	The *role* attribute addresses the role of the full set of contact information with regards to the entity defined by the concept. Examples: "privateOffice" vs "companyOffice" or "GlobalHeadquarters" vs "localHeadquarterUK".
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the "role" of an event's contact information: http://cv.iptc.org/newscodes/eventcontactinforole/
Attribute(s)	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the set of contact information.
Child Element(s)	▪ **email** (page 103) (0..unbounded) ▪ **im** (page 132) (0..unbounded) ▪ **phone** (page 164) (0..unbounded) ▪ **fax** (page 113) (0..unbounded) ▪ **web** (page 201) (0..unbounded) ▪ **address** (page 166) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.27 Content Metadata for Concept Items

Table 34. Content Metadata for Concept Items

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta		
Title	Content Metadata for Concept Items		
Definition	Content Metadata for a Concept Item		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 243)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **Administrative Metadata Group** (page 203) (0..1)	**Element Name**	**Page**
		audience (0..unbounded)	56
		contributor (0..unbounded)	80
		creator (0..unbounded)	85
		contentCreated (0..1)	86
		contentModified (0..1)	87
		located (0..unbounded)	144
		infoSource (0..unbounded)	133
		urgency (0..1)	198
		exclAudience (0..unbounded)	109
		altId (0..unbounded)	52
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

11.6.28 Content Metadata for Knowledge Items

Table 35. Content Metadata for Knowledge Items

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta		
Title	Content Metadata for Knowledge Items		
Definition	Content Metadata for a Knowledge Item.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 243)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **Administrative Metadata Group** (page 203) (0..1)	**Element Name**	**Page**
		audience (0..1)	56
		contributor (0..unbounded)	80
		creator (0..unbounded)	85
		contentCreated (0..1)	86
		contentModified (0..1)	87
		located (0..unbounded)	144
		infoSource (0..unbounded)	133
		urgency (0..1)	198
		exclAudience (0..unbounded)	109
		altId (0..unbounded)	52
	▪ **Knowledge Descriptive Metadata Group** (page 203) (0..1)	**Element Name**	**Page**
		description (0..unbounded)	98
		subject (0..unbounded)	193
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

11.6.29 Content Metadata for News Items

Table 36. Content Metadata for News Items

(XML) Data Model	Element			
Namespace (prefix)	nar			
Name	contentMeta			
Title	Content Metadata for News Items			
Definition	A set of properties about the content.			
User Note(s)				
Implementation Note(s)				
XML Schema Spec	At: PCL			
Datatype				
Internally Ctrl Values				
Externally Ctrl Values				
Attribute(s)	▪ **i18nAttributes** (page 243)		Name	Datatype
			xml:lang (0..1)	XML Schema language
			dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **Administrative Metadata Group** (page 203) (0..1)		**Element Name**	**Page**
			audience (0..unbounded)	56
			contributor (0..unbounded)	80
			creator (0..unbounded)	85
			contentCreated (0..1)	86
			contentModified (0..1)	87
			located (0..unbounded)	144
			infoSource (0..unbounded)	133
			urgency (0..1)	198
			exclAudience (0..unbounded)	109
			altId (0..unbounded)	52
	▪ **Descriptive Metadata Group** (page 204) (0..1)		**Element Name**	**Page**
			by (0..unbounded)	60
			dateline (0..unbounded)	96
			description (0..unbounded)	98
			genre (0..unbounded)	116
			headline (0..unbounded)	122
			language (0..unbounded)	140
			slugline (0..unbounded)	190
			subject (0..unbounded)	193
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.			
XML Schema Note(s)				
Example(s)				

11.6.30 Content Metadata for Package Items

Table 37. Content Metadata for Package Items

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta		
Title	Content Metadata for Package Items		
Definition	A set of properties about the content.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 243)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **Administrative Metadata Group** (page 203) (0..1)	**Element Name**	**Page**
		audience (0..unbounded)	56
		contributor (0..unbounded)	80
		creator (0..unbounded)	85
		contentCreated (0..1)	86
		contentModified (0..1)	87
		located (0..unbounded)	144
		infoSource (0..unbounded)	133
		urgency (0..1)	198
		exclAudience (0..unbounded)	109
		altId (0..unbounded)	52
	▪ **Descriptive Metadata Group** (page 204) (0..1)	**Element Name**	**Page**
		by (0..unbounded)	60
		dateline (0..unbounded)	96
		description (0..unbounded)	98
		genre (0..unbounded)	116
		headline (0..unbounded)	122
		language (0..unbounded)	140
		slugline (0..unbounded)	190
		subject (0..unbounded)	193
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

11.6.31 Content Provider

Table 38. Content Provider

(XML) Data Model	Element
Namespace (prefix)	nar
Name	provider
Title	Provider
Definition	The party responsible for the management and the release of the Item.
User Note(s)	This corresponds to the publisher of the Item.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**FlexPartyPropType** (page 222)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.32 Content Set

Table 39. Content Set

(XML) Data Model	Element
Namespace (prefix)	nar
Name	contentSet
Title	Content Set
Definition	A set of alternate renditions of the Item content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ original (0..1); XML Schema idref; A local reference to the original piece of content, from which all renditions have been derived.
Child Element(s)	▪ **inlineXML** (page 129) (0..unbounded) or ▪ **inlineData** (page 125) (0..unbounded) or ▪ **remoteContent** (page 177) (0..unbounded)
XML Schema Note(s)	
Example(s)	

11.6.33 Contributor

Table 40. Contributor

(XML) Data Model	Element
Namespace (prefix)	nar
Name	contributor
Title	Contributor
Definition	A party (person or organisation) which modified or enhanced the content, preferably the name of a person.
User Note(s)	One may specify the role the party plays in the creation of the content (e.g. a caption writer for photos) at the PCL.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PartyPropType** (page 217)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.34 Copyright Holder

Table 41. Copyright Holder

(XML) Data Model	Element
Namespace (prefix)	nar
Name	copyrightHolder
Title	Copyright Holder
Definition	The person or organisation claiming the intellectual property for the content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**FlexPartyPropType** (page 222)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.35 Copyright Notice

Table 42. Copyright Notice

(XML) Data Model	Element
Namespace (prefix)	nar
Name	copyrightNotice
Title	Copyright Notice
Definition	Any necessary copyright notice for claiming the intellectual property for the content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**RightsLabelType** (page 238)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.36 Country

Table 43. Country

(XML) Data Model	Element
Namespace (prefix)	nar
Name	country
Title	Country
Definition	A country, part of the address.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.37 Country Area

Table 44. Country Area

(XML) Data Model	Element
Namespace (prefix)	nar
Name	area
Title	Country Area
Definition	A subdivision of a country, part of the address.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.38 Creator

Table 45. Creator

(XML) Data Model	Element
Namespace (prefix)	nar
Name	creator
Title	Creator
Definition	A party (person or organisation) which created the content, preferably the name of a person (e.g. a photographer for photos, a graphic artist for graphics, or a writer for textual news).
User Note(s)	One may specify the role the party plays in the creation of the content (e.g. a caption writer for photos) at the PCL.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PartyPropType** (page 217)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.39 Date Content Created

Table 46. Date Content Created

(XML) Data Model	Element
Namespace (prefix)	nar
Name	contentCreated
Title	Date Content Created
Definition	The date (and optionally the time with the time zone) at which the content was created.
User Note(s)	In the case of a photo or live footage for audio and video, this date (and time) is always the same as the date (and time) of the event covered by the content. In the case of text and any audio and video report about an event, this date (and time) can be different from the date (and time) of the event covered by the content. This date (and time) may also be different from the date (and time) of the creation of an Item holding the content.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 239)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.40 Date Content Modified

Table 47. Date Content Modified

(XML) Data Model	Element
Namespace (prefix)	nar
Name	contentModified
Title	Date Content Modified
Definition	The date (and optionally the time with the time zone) at which the content was last modified.
User Note(s)	The value of this property should be updated each time the content is modified in any manner, but should not be updated if only metadata are changed.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 239)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.41 Date Item Embargo Ends

Table 48. Date Item Embargo Ends

(XML) Data Model	Element
Namespace (prefix)	nar
Name	embargoed
Title	Date Item Embargo Ends
Definition	The date and time (with the time zone) before which all versions of the Item are embargoed. If absent, the Item is not embargoed.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**DateTimePropType** (page 213)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.42 Date Item First Created

Table 49. Date Item First Created

(XML) Data Model	Element
Namespace (prefix)	nar
Name	firstCreated
Title	Date Item First Created
Definition	The date and time (with the time zone) on which the first version of the Item was created.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**DateTimePropType** (page 213)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.43 Date Item Version Created

Table 50. Date Item Version Created

(XML) Data Model	Element
Namespace (prefix)	nar
Name	versionCreated
Title	Date Item Version Created
Definition	The date and time (with the time zone) on which the current version of the Item was created.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**DateTimePropType** (page 213)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.44 Date of Birth

Table 51. Date of Birth

(XML) Data Model	Element
Namespace (prefix)	nar
Name	born
Title	Date of Birth
Definition	The date of birth of the person.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 239)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.45 Date of Death

Table 52. Date of Death

(XML) Data Model	Element
Namespace (prefix)	nar
Name	died
Title	Date of Death
Definition	The date of death of the person.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 239)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.46 Date of Dissolution

Table 53. Date of Dissolution

(XML) Data Model	Element
Namespace (prefix)	nar
Name	dissolved
Title	Date of Dissolution
Definition	The date of dissolution of the organisation.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 239)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.47 Date of Foundation

Table 54. Date of Foundation

(XML) Data Model	Element
Namespace (prefix)	nar
Name	founded
Title	Date of Foundation
Definition	The date of foundation/establishment of the organisation.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 239)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.48 Date of Transmission

Table 55. Date of Transmission

(XML) Data Model	Element
Namespace (prefix)	nar
Name	sent
Title	Date of Transmission
Definition	The date and time (with the time zone) of transmission of the message.
User Note(s)	May not be updated in case of retransmission of the message.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema dateTime
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.49 Dateline

Table 56. Dateline

(XML) Data Model	Element
Namespace (prefix)	nar
Name	dateline
Title	Dateline
Definition	A natural-language statement of the date and/or place of creation of the content.
User Note(s)	The dateline provides a natural-language statement of the date and/or place of the news content creation, to be displayed in situations where an abstract of the content is shown (case of search results) or the content is remote. Traditionally a dateline indicates when and where news content is created, not necessarily the time and place relative to the news event. As an example a dateline BAGHDAD, March 26, 2007 (AFP) could head a story about blast in Mosul, because the story was actually written in Baghdad. Also, by tradition a dateline will follow the stylebook of the information provider and possibly leave out certain time and location information that could be useful for specifying searches of a database. Editorial policy dictates the dateline; it is not automatically derivable from other markup (location, date, etc.). The dateline should not end with a separating character (of the kind that separates the dateline from the first sentence in a traditional wire story).
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Label1Type** (page 229)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.50 Dates

Table 57. Dates

(XML) Data Model	Element
Namespace (prefix)	nar
Name	dates
Title	Dates
Definition	All dates pertaining to the event, in particular the start and end date and any recurrence information.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **start** (page 192) (1)

For the Child Element(s) row, the following sub-content appears:

▪ **start** (page 192) (1)

▪ **end** (page 104) (0..1)
or
▪ **duration** (page 100) (0..1)

▪ **confirmation** (page 72) (0..1)

▪ **Recurrence Group** (page 202) (0..1)

Element Name	Page
exDate (1..unbounded)	110
exRule (1..unbounded)	111
rDate (0..unbounded)	172
rRule (0...unbounded)	173

XML Schema Note(s)	
Example(s)	

11.6.51 Description

Table 58. Description

(XML) Data Model	Element
Namespace (prefix)	nar
Name	description
Title	Description
Definition	A free-form textual description of the content of the item. (For a Knowledge Item the content is its set of concepts as a whole.)
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**BlockType** (page 208)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the *role* attribute: http://cv.iptc.org/newscodes/descriptionrole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.52 Destination

Table 59. Destination

(XML) Data Model	Element
Namespace (prefix)	nar
Name	destination
Title	Destination
Definition	The point(s) of destination of the message.
User Note(s)	In a broadcast delivery system, the destination is a group of reception points (using a provider-specific syntax, often geographically oriented). This is a way to address customers. Examples are "England", "USA", "Austria/Vienna", "France/Paris/LeParisien". The structure of this string is not specified by the IPTC.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.53 Duration

Table 60. Duration

(XML) Data Model	Element
Namespace (prefix)	nar
Name	duration
Title	Duration
Definition	The period the event will last. The duration is calculated from the date and time of the **start** (page 192) property.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	xs:duration
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.54 Editorial Note

Table 61. Editorial Note

(XML) Data Model	Element
Namespace (prefix)	nar
Name	edNote
Title	Editorial Note
Definition	A note addressed to the editorial people receiving and processing the Item.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**BlockType** (page 208)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.55 Editorial Service

Table 62. Editorial Service

(XML) Data Model	Element
Namespace (prefix)	nar
Name	service
Title	Editorial Service
Definition	An editorial service to which an Item is assigned to by its provider.
User Note(s)	The values of this property are defined by each provider, and are often associated with the notion of a desk or a feed. Some examples are a "French wire service", an "international picture service" or a "mobile news service".
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QualPropType** (page 235)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.56 Email Address

Table 63. Email Address

(XML) Data Model	Element
Namespace (prefix)	nar
Name	email
Title	Email Address
Definition	An email address.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**ElectronicAddressType** (page 214)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.57 End Date/Time

Table 64. End Date/Time

(XML) Data Model	Element
Namespace (prefix)	nar
Name	end
Title	End Date/Time
Definition	The date (and optionally the time with the time zone) the event ends. This may be an exact or an approximative value.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**ApproximateDateTimePropType** (page 206)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.58 Event

Table 65. Event

(XML) Data Model	Element
Namespace (prefix)	nar
Name	event
Title	Event
Definition	Structured information about an event without a concept identifier.
User Note(s)	
Implementation Note(s)	This event structure is used within an events wrapper to be plugged into an inlineXML property of a News Item.
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **eventDetails** (page 106) (1)

	Element Name	Page
▪ **name** (page 70) (1..unbounded)		
▪ **definition** (page 67) (0..unbounded)		
▪ **facet** (page 112) (0..unbounded)		
▪ **note** (page 153) (0..unbounded)		
▪ **Concept Relationships Group** (page 202) (1)	**broader** (0..unbounded)	59
	narrower (0..unbounded)	149
	related (0..unbounded)	175
	sameAs (0..unbounded)	186

XML Schema Note(s)	
Example(s)	

11.6.59 Event Details

Table 66. Event Details

(XML) Data Model	Element
Namespace (prefix)	nar
Name	eventDetails
Title	Event Details
Definition	Details about the event.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **dates** (page 97) (1)
	▪ **occurStatus** (page 154) (0..1)
	▪ **registration** (page 174) (0..1)
	▪ **accessStatus** (page 48) (0..unbounded)
	▪ **subject** (page 193) (0..unbounded);
	▪ **location** (page 107) (0..unbounded)
	▪ **participant** (page 160) (0..unbounded)
	▪ **participationRequirement** (page 161) (0..unbounded)
	▪ **organiser** (page 158) (0..unbounded)
	▪ **contactInfo** (page 73) (0..unbounded)
	▪ **language** (page 140) (0..unbounded)
	▪ **newsCoverage** (page 150) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.60 Event Location

Table 67. Event Location

(XML) Data Model	Element
Namespace (prefix)	nar
Name	location
Title	Event Location
Definition	A location (geographical area or point of interest) in which the event takes place.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**FlexLocationPropType** (page 219)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeType** (page 234); A refinement on the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.61 Events Wrapper

Table 68. Events Wrapper

(XML) Data Model	Element
Namespace (prefix)	nar
Name	events
Title	Events Wrapper
Definition	A wrapper for events in a News Item.
User Note(s)	
Implementation Note(s)	This events wrapper is made to be plugged into an inlineXML property of a News Item.
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **Event** (page 105) (1..unbounded)
XML Schema Note(s)	
Example(s)	

11.6.62 Excluded Audience

Table 69. Excluded Audience

(XML) Data Model	Element
Namespace (prefix)	nar
Name	exclAudience
Title	Excluded Audience
Definition	An excluded audience for the content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**AudienceType** (page 207)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.63 Exclusion Date

Table 70. Exclusion Date

(XML) Data Model	Element
Namespace (prefix)	nar
Name	exDate
Title	Exclusion Date
Definition	An explicit date (and optionally time with the time zone) to be excluded from the recurrence set.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**DateOptTimePropType** (page 211)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.64 Exclusion Rule

Table 71. Exclusion Rule

(XML) Data Model	Element
Namespace (prefix)	nar
Name	exRule
Title	Exclusion Rule
Definition	A rule of recurrence exclusion.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**RecurrenceRuleType** (page 236)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.65 Facet

Table 72. Facet

(XML) Data Model	Element
Namespace (prefix)	nar
Name	facet
Title	Facet
Definition	An intrinsic property of the concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**TypedQualPropType** (page 241)
Internally Ctrl Values	
Externally Ctrl Values	The default value and additional values for the rel attribute are defined by the IPTC Facet Relationship NewsCodes - http://cv.iptc.org/newscodes/facetrelation/
Attribute(s)	▪ rel (0..1); **QCodeType** (page 234); The identifier of the relationship between the current concept (containing the facet) and the concept identified by the facet value. The default value for *rel* is the "IsA" relationship, this applies also if the *rel* attribute is omitted. <table><tr><th></th><th>Name</th><th>DataType</th></tr><tr><td rowspan="2">▪ **timeValidityAttributes** (page 244)</td><td>validfrom (0..1)</td><td>DateOptTimeType</td></tr><tr><td>validto (0..1)</td><td>DateOptTimeType</td></tr></table>
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.66 Fax Number

Table 73. Fax Number

(XML) Data Model	Element
Namespace (prefix)	nar
Name	fax
Title	Fax Number
Definition	An international fax number.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**ElectronicAddressType** (page 214)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.67 File Name

Table 74. File Name

(XML) Data Model	Element
Namespace (prefix)	nar
Name	filename
Title	File Name
Definition	The recommended file name for this Item.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML Schema normalizedString
Internally Ctrl Values	
Externally Ctrl Values	

Attribute(s)	▪ **editAttributes** (page 243)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType

Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.68 Generator Tool

Table 75. Generator Tool

(XML) Data Model	Element
Namespace (prefix)	nar
Name	generator
Title	Generator Tool
Definition	The name and version of the software tool used to generate the Item.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**VersionedStringType** (page 242)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.69 Genre

Table 76. Genre

(XML) Data Model	Element
Namespace (prefix)	nar
Name	genre
Title	Genre
Definition	A nature, intellectual or journalistic form of the news content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1ConceptPropType** (page 216)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.70 Geographic Position

Table 77. Geographic Position

(XML) Data Model	Element
Namespace (prefix)	nar
Name	position
Title	Geographic Position
Definition	The geographic coordinates of the location.
User Note(s)	These properties follow the syntax used by the major geocoders on the Web. Latitudes north of the equator shall be designated by use of the plus sign (+), latitudes south of the equator shall be designated by use of the minus sign (-). The equator shall be designated by use of the plus sign (+). Longitudes east of Greenwich shall be designated by use of the plus sign (+), longitudes west of Greenwich shall be designated by use of the minus sign (-). The Prime Meridian shall be designated by use of the plus sign (+). The 180th meridian shall be designated by use of the minus sign (-). The altitude is given in meters. A positive integer means a position above the zero elevation, a negative value below the zero elevation. In the absence of the *gpsdatum* attribute, WGS84 is the default system.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ latitude (1); XML Schema decimal; The latitude in decimal degrees (Positive value = northern latitude, negative value = southern latitude).
	▪ longitude (1); XML Schema decimal; The longitude in decimal degrees (Positive value = eastern longitude, negative value = western longitude).
	▪ altitude (0..1); XML Schema integer; The altitude in meters above the zero elevation of the reference system (sea level).
	▪ gpsdatum (0..1); XML Schema string; The GPS datum associated with the measure.
Child Element(s)	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.71 Geopolitical Area Details

Table 78. Geopolitical Area Details

(XML) Data Model	Element
Namespace (prefix)	nar
Name	geoAreaDetails
Title	Geopolitical Area Details
Definition	A set of properties specific for a geopolitical area.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **position** (page 117) (0..1)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.72 Group

Table 79. Group

(XML) Data Model	Element
Namespace (prefix)	nar
Name	group
Title	Group
Definition	A mixed set of group references and links.
User Note(s)	▪ Group Mode: By default the group is "complementary and unordered". 　▪ Complementary and Unordered: To be used for any kind of supporting content that does not require a sequence to be specified. 　▪ Complementary and Ordered: The group starts with the first child of the group. To be used for any kind of content which must be displayed or consumed in a particular sequence, expressed by the order of the child elements of the group. The semantics of the role attribute value determine the required processing. 　▪ Alternatives: To be used if a group contains equivalent pieces of content (e.g. translations of the same news story into different languages). The recipient may pick one or more of these. ▪ Group References and Item References: Can be included in any order, and this order may be relevant or not, depending the value of the mode attribute. Each link aggregates an external resource (Item or Web resource) to the package. Optionally, it indicates the relationship between the group and the target resource plus some additional hints about the resource itself.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (1); **QCodeType** (page 234); The part this group plays within its container.
	▪ mode (0..unbounded); **QCodeType** (page 234); An indication whether the elements in the group are complementary and unordered, complementary and ordered or a set of alternative elements.
	▪ id (1); XML Schema id; The local identifier of the group.
	▪ creator (0..1); QCodeType; party who has edited the property (or will edit the property)
	▪ modified (0..1); DateOptTimeType; The date (and optionally the time) when the property was last modified.

	Name	Datatype
▪ **i18nAttributes** (page 243)	xml:lang (0..1)	XML Schema language
	dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	▪ **groupRef** (page 120) (0..unbounded)
	▪ **itemRef** (page 136) (0..unbounded)
XML Schema Note(s)	The local identifier (id) common to all elements at PCL provides a local identifier for groups.
Example(s)	

11.6.73 Group Reference

Table 80. Group Reference

(XML) Data Model	Element
Namespace (prefix)	nar
Name	groupRef
Title	Group Reference
Definition	A reference to a group local to the package.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ idref (1); XML Schema idref; The reference to the id of a local group.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.74 Group Set

Table 81. Group Set

(XML) Data Model	Element
Namespace (prefix)	nar
Name	groupSet
Title	Group Set
Definition	A hierarchical set of groups.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ root (1); XML Schema idref; The reference to a local group acting as the root of the hierarchy of groups.
Child Element(s)	▪ **group** (page 119) (1..unbounded)
XML Schema Note(s)	
Example(s)	

11.6.75 Headline

Table 82. Headline

(XML) Data Model	Element
Namespace (prefix)	nar
Name	headline
Title	Headline
Definition	A brief and snappy introduction to the news content, designed to catch the reader's attention.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Label1Type** (page 229)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.76 Icon

Table 83. Icon

(XML) Data Model	Element
Namespace (prefix)	nar
Name	icon
Title	Icon
Definition	A small image illustrating (a part of) the content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ href (1); **IRIType** (page 228); The URL of the image acting as an icon for the part.

Attribute(s) (cont.)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType

Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.77 Inline Concept Marker

Table 84. Inline Concept Marker

(XML) Data Model	Element
Namespace (prefix)	nar
Name	inline
Title	Inline Concept Marker
Definition	An inline markup tag to be used with any concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Mixed Content
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ class (0..1); XML Schema String; An equivalent of the HTML *class* attribute.
	▪ qcode (0..1); **QCodeType** (page 234); A qualified code assigned as property value. or
	▪ literal (0..1); XML Schema normalizedString; A free-text value assigned as property value.
	▪ type (0..1); **QCodeType** (page 234); The type of the concept assigned as controlled or uncontrolled property value.

Attribute(s) nested tables:

▪ **i18nAttributes** (page 243)

Name	Datatype
xml:lang (0..1)	XML Schema language
dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

▪ **editAttributes** (page 243)

Name	Datatype
id (0..1)	XML Schema ID
creator (0..1)	QCodeType
modified (0..1)	DateOptTimeType

▪ **quantifyAttributes** (page 243)

Name	Datatype
confidence (0..1)	Int100Type
relevance (0..1)	Int100Type
why (0..1)	QCodeType

Child Element(s)	▪ **span** (page 191) (0..unbounded)
	▪ **ruby** (page 182) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.78 Inline Data (NewsML-G2 Specific)

Table 85. Inline Data (NewsML-G2 Specific)

(XML) Data Model	Element
Namespace (prefix)	nar
Name	inlineData
Title	Inline Data (NewsML-G2 Specific)
Definition	The encoding applied to the content before inclusion.
User Note(s)	
Implementation Note(s)	For the encoding attribute at the CCL only the QCode for "base64" may be used. If the attribute does not exist, this QCode must be assumed as default.. In the absence of the encoding attribute, the content must be plain text, and the content type must be set accordingly.
XML Schema Spec	At PCL
Datatype	XML schema string
Internally Ctrl Values	
Externally Ctrl Values	

Table 85. Inline Data (NewsML-G2 Specific) (Continued)

Attribute(s)		Name	Datatype
	▪ encoding (0..1); **QCodeType** (page 234); Specifies the encoding applied to the content before inclusion in the content.		
	▪ i18nAttributes (page 243)	xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
	▪ newsContentAttributes (page 244)	id (0..1)	XML Schema ID
		rendition (0..1)	QCodeType
		contenttype (0..1)	XML Schema string
		format (0..1)	QCodeType
		generator (0..1)	XML Schema string
		generated (0..1)	DateOptTimeType
	▪ newsContentCharacteristics (page 245)	wordcount (0..1)	XML Schema nonNegativeInteger
		width (0..1)	XML Schema nonNegativeInteger
		height (0..1)	XML Schema nonNegativeInteger
		orientation (0..1)	XML Schema nonNegativeInteger
		colourspace (0..1)	QCodeType
		resolution (0..1)	XML Schema positiveInteger
		duration (0..1)	XML Schema nonNegativeInteger
		audiocodec (0..1)	XML Schema nomalizedString
		audiobitrate (0..1)	XML Schema positiveInteger
		audiovbr (0..1)	enumeration: yes/no
		audiosamplesize (0..1)	XML Schema positiveInteger
		audiosamplerate (0..1)	XML Schema positiveInteger
		audiochannels (0..1)	QCodeType
		videocodec (0..1)	XML Schema nomalizedString
		videoavgbitrate (0..1)	XML Schema positiveInteger
		videovbr (0..1)	enumeration: yes/no
		videoframerate (0..1)	XML Schema positiveInteger
Child Element(s)			

Table 85. Inline Data (NewsML-G2 Specific) (Continued)

XML Schema Note(s)	
Example(s)	

11.6.79 Inline Reference

Table 86. Inline Reference

(XML) Data Model	Element
Namespace (prefix)	nar
Name	inlineRef
Title	Inline Reference
Definition	A concept represented by the content identified by the local identifier(s).
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ idrefs (0..1); XML Schema IDREFS; A set of local identifiers of inline content.

Attribute(s) — ▪ **quantifyAttributes** (page 243):

Name	Datatype
confidence (0..1)	Int100Type
relevance (0..1)	Int100Type
why (0..1)	QCodeType

Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.80 Inline XML (NewsML-G2 Specific)

Table 87. Inline XML (NewsML-G2 Specific)

(XML) Data Model	Element
Namespace (prefix)	nar
Name	inlineXML
Title	Inline XML (NewsML-G2 Specific)
Definition	A rendition of the content using an XML language.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the *encoding* attribute: http://cv.iptc.org/newscodes/encoding/

Table 87. Inline XML (NewsML-G2 Specific) (Continued)

<table>
<tr><td rowspan="28">Attribute(s)</td><td rowspan="2">▪ i18nAttributes (page 243)</td><td>Name</td><td>Datatype</td></tr>
<tr><td>xml:lang (0..1)</td><td>XML Schema language</td></tr>
<tr><td></td><td>dir (0..1)</td><td>XML Schema string: enumeration <i>ltr</i>, <i>rtl</i>.</td></tr>
<tr><td rowspan="7">▪ newsContentAttributes (page 244)</td><td>Name</td><td>Datatype</td></tr>
<tr><td>id (0..1)</td><td>XML Schema ID</td></tr>
<tr><td>rendition (0..1)</td><td>QCodeType</td></tr>
<tr><td>contenttype (0..1)</td><td>XML Schema string</td></tr>
<tr><td>format (0..1)</td><td>QCodeType</td></tr>
<tr><td>generator (0..1)</td><td>XML Schema string</td></tr>
<tr><td>generated (0..1)</td><td>DateOptTimeType</td></tr>
<tr><td rowspan="19">▪ newsContentCharacteristics (page 245)</td><td>Name</td><td>Datatype</td></tr>
<tr><td>wordcount (0..1)</td><td>XML Schema nonNegativeInteger</td></tr>
<tr><td>width (0..1)</td><td>XML Schema nonNegativeInteger</td></tr>
<tr><td>height (0..1)</td><td>XML Schema nonNegativeInteger</td></tr>
<tr><td>orientation (0..1)</td><td>XML Schema nonNegativeInteger</td></tr>
<tr><td>colourspace (0..1)</td><td>QCodeType</td></tr>
<tr><td>resolution (0..1)</td><td>XML Schema positiveInteger</td></tr>
<tr><td>duration (0..1)</td><td>XML Schema nonNegativeInteger</td></tr>
<tr><td>audiocodec (0..1)</td><td>XML Schema nomalizedString</td></tr>
<tr><td>audiobitrate (0..1)</td><td>XML Schema positiveInteger</td></tr>
<tr><td>audiovbr (0..1)</td><td>enumeration: yes/no</td></tr>
<tr><td>audiosamplesize (0..1)</td><td>XML Schema positiveInteger</td></tr>
<tr><td>audiosamplerate (0..1)</td><td>XML Schema positiveInteger</td></tr>
<tr><td>audiochannels (0..1)</td><td>QCodeType</td></tr>
<tr><td>videocodec (0..1)</td><td>XML Schema nomalizedString</td></tr>
<tr><td>videoavgbitrate (0..1)</td><td>XML Schema positiveInteger</td></tr>
<tr><td>videovbr (0..1)</td><td>enumeration: yes/no</td></tr>
<tr><td>videoframerate (0..1)</td><td>XML Schema positiveInteger</td></tr>
<tr><td>Child Element(s)</td><td colspan="3">▪ Plug-in Point (0..1). XML content from any namespace.</td></tr>
<tr><td>XML Schema Note(s)</td><td colspan="3"></td></tr>
<tr><td>Example(s)</td><td colspan="3"></td></tr>
</table>

11.6.81 Instance Of

Table 88. Instance Of

(XML) Data Model	Element
Namespace (prefix)	nar
Name	instanceOf
Title	Instance Of
Definition	A frequently updated information object of which this Item is an instance.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.82 Instant Messaging Address

Table 89. Instant Messaging Address

(XML) Data Model	Element
Namespace (prefix)	nar
Name	im
Title	Instant Messaging Address
Definition	An address of an instant messaging system.
User Note(s)	The tech attribute indicates the provider of the service (Yahoo!, Google etc.).
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**ElectronicAddressTechType** (page 215)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.83 Information Source

Table 90. Information Source

(XML) Data Model	Element
Namespace (prefix)	nar
Name	infoSource
Title	Information Source
Definition	A party (person or organisation) which originated some information used to create or enhance the content.
User Note(s)	If an entity plays more than one role, the *infoSource* element has to be included multiple times, with different values of role.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PartyPropType** (page 217)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the *role* attribute: http://cv.iptc.org/newscodes/infosourcerole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.84 Item Class

Table 91. Item Class

(XML) Data Model	Element
Namespace (prefix)	nar
Name	itemClass
Title	Item Class
Definition	Indicates the nature of the Item.
User Note(s)	This property gives a hint on the nature of the Item. IPTC values for News Items correspond to the media type of the original content component, i.e. "text", "photo", etc. Concept Items adopt the static value *concept*. The class of a Package Item reflects the nature of the items it contains, i.e. either one of the values above or the value "composite" which indicates that the package handles items of different natures. A recipient system may use this information to make a coarse selection of Items, based on their nature, without having to inspect the structure.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QualPropType** (page 235)
Internally Ctrl Values	
Externally Ctrl Values	Mandatory IPTC NewsCodes for News Items or Package Items: http://cv.iptc.org/newscodes/ninature/ Mandatory IPTC NewsCodes for Concept Items, Knowledge Items or Package Items: http://cv.iptc.org/newscodes/cinature/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.85 Item Metadata

Table 92. Item Metadata

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	itemMeta		
Title	Item Metadata		
Definition	A set of properties directly associated with the Item.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)			
Child Element(s)	▪ **Item Management Group** (page 204) (1)	**Element Name**	**Page**
		provider (1)	78
		embargoed (0..1)	88
		firstCreated (0..1)	89
		versionCreated (1)	90
		edNote (0..unbounded)	101
		service (0..unbounded)	102
		filename (0..1)	114
		itemClass (1)	134
		pubStatus (0..1)	171
		role (0..1)	181
		title (0..unbounded)	138
		altRep (0..unbounded)	53
		generator (0..1)	115
		instanceOf (0..unbounded)	131
		memberOf (0..unbounded)	147
		profile (0..1)	169
		signal (0..unbounded)	189
	▪ **link** (page 142) (0..unbounded)		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

11.6.86 Item Reference

Table 93. Item Reference

(XML) Data Model	Element
Namespace (prefix)	nar
Name	itemRef
Title	Item Reference
Definition	A reference to a target Item or Web resource.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Link1Type** (page 230)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.87 Item Set

Table 94. Item Set

(XML) Data Model	Element
Namespace (prefix)	nar
Name	itemSet
Title	Item Set
Definition	A set of Items.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **newsItem** (page 151) (1..unbounded) ▪ **conceptItem** (page 69) (1..unbounded) ▪ **packageItem** (page 159) (1..unbounded) ▪ **knowledgeItem** (page 139) (0..unbounded)
XML Schema Note(s)	To allow the validation of the structure beyond the root elements of the different items the extension point "any" for the nar XML namespace is the only child element. This allows schema based validation of the content of the items as the validation of the extension point is set to "lax".
Example(s)	

11.6.88 Item Title

Table 95. Item Title

(XML) Data Model	Element
Namespace (prefix)	nar
Name	title
Title	Item Title
Definition	A short, natural-language name for the Item.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Label1Type** (page 229)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.89 Knowledge Item

Table 96. Knowledge Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	knowledgeItem
Title	Knowledge Item
Definition	An Item used for collating a set of concept definitions to form the physical representation of a controlled vocabulary.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta** (page 75) (0..1)
	▪ **assert** (page 55) (0..unbounded)
	▪ **inlineRef** (page 128) (0..unbounded)
	▪ **conceptSet** (page 71) (0..1)
XML Schema Note(s)	
Example(s)	

11.6.90 Language

Table 97. Language

(XML) Data Model	Element
Namespace (prefix)	nar
Name	language
Title	Language
Definition	A language associated with the content. For news this is a language used by the news content, for events this is a language used at this event, for Knowledge Items this is the major language used to describe the concepts
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	*tag* values MUST conform to BCP47. Recommended IPTC NewsCodes for the *role* attribute: http://cv.iptc.org/newscodes/languagerole/
Attribute(s)	▪ tag (1); XML Schema language; Indicator of the language.
	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the property.

Attribute(s)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType

Child Element(s)	▪ **name** (page 170) (0..unbounded)
XML Schema Note(s)	
Example(s)	

11.6.91 Line Break

Table 98. Line Break

(XML) Data Model	Element
Namespace (prefix)	nar
Name	br
Title	Line Break
Definition	A line break.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	Empty element
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.92 Link

Table 99. Link

(XML) Data Model	Element
Namespace (prefix)	nar
Name	link
Title	Link
Definition	A link from the current Item to a target Item or Web resource.
User Note(s)	They are different variants of links: Links may allow for navigation from a newsItem to another related Item or a Web resource, and its title be displayed as supplemental information to the final user. Example: a newsItem representing a section of a transcript (a "take" in the news language) may be linked to the previous and next take; an article about a person may be linked to the biography of this person. Links may express a parent-child relationship. Example: a newsItem representing an article may be linked to the article it is a translation of; a wrap-up may be linked to the previous stories used as source material for the article; a cropped picture may be linked to its source picture. Links may express dependency on external Items which are required in order to fully present the composite content of the Item. If some target Items are not retrievable, then the recipient processor should fail gracefully. The most obvious example is a newsItem representing an illustrated article. The textual content of the newsItem (usually formatted as NITF or XHTML) includes a reference to a photo which is represented by another newsItem. As the NAR recipient processor is content agnostic, it cannot infer this dependency from processing the content. A dependency link from the article to the picture indicates that the recipient processor must retrieve the target newsItem before the article can be fully displayed. Pointing at the latest version of an Item while exposing content metadata may lead to unwanted display or selection criteria if these metadata were subsequently modified; therefore only the stable content properties should be exposed in a link.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Link1Type** (page 230)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.93 Locality

Table 100. Locality

(XML) Data Model	Element
Namespace (prefix)	nar
Name	locality
Title	Locality
Definition	A city/town/village etc. part of the address.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.94 Located

Table 101. Located

(XML) Data Model	Element
Namespace (prefix)	nar
Name	located
Title	Located
Definition	A location from which the content originates.
User Note(s)	This information applies especially to news, and may also be expressed as free text in the "dateline" of a story, along with a date of content creation and the name of the content provider. The rules for determining the location are provider-dependent. The location is typically determined differently for different types of content: - Text: The practices of news providers either identify the location the content relates to or the location the content was created by a reporter or a writer. If a correspondent is resident in town A but writes about an event in town B the name of town A or B can be used. But the provider's policy should be available as written document. - Photo: The location of origin of content is the place shown in the photo image. - Graphics: The location of origin of content should be the editorial office from where this graphics are distributed. - Audio and video: In the case of raw footage the location of origin of the content should be the place of event, if people can be heard/are shown from different places the news provider can decide by its own policy, but this policy should be available as written document.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**FlexLocationPropType** (page 219)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.95 Location

Table 102. Location

(XML) Data Model	Element
Namespace (prefix)	nar
Name	location
Title	Location
Definition	A location (geographical area or point of interest).
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**FlexLocationPropType** (page 219)
Internally Ctrl Values	
Externally Ctrl Values	

		Name	DataType
Attribute(s)	▪ **timeValidityAttributes** (page 244)	validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType

Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.96 Location Details

Table 103. Location Details

(XML) Data Model	Element
Namespace (prefix)	nar
Name	details
Title	Location Details
Definition	Detailed information about the precise location of the point of interest.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**BlockType** (page 208)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.97 Member Of

Table 104. Member Of

(XML) Data Model	Element
Namespace (prefix)	nar
Name	memberOf
Title	Member Of
Definition	A set of Items around the same theme of which this Item is a part.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.98 Message Header

Table 105. Message Header

(XML) Data Model	Element
Namespace (prefix)	nar
Name	header
Title	Message Header
Definition	A set of properties facilitating the exchange of Items.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **sent** (page 95) (1); May not be updated in case message retransmission. ▪ **sender** (page 188) (0..1); The structure of this string is not specified by the IPTC. Best practice is to identify a sender by its domain name. ▪ **transmitId** (page 196) (0..1); No two News Messages sent by the same sender on the same date can have the same identifier. In case of retransmission it is not required to update this identifier. This string structure is not specified by the IPTC. ▪ **priority** (page 168) (0..1) ▪ **origin** (page 156) (0..1); This string structure is not specified by the IPTC. ▪ **destination** (page 99) (0..unbounded) ▪ **channel** (page 65) (0..unbounded); A channel identifier is used to provide recipients with information on which select, route, or otherwise handle the content of the message. The channels represent streams in a multiplex: a message may be sent on different channels – e.g. one for text, one for pictures – and each reception point will be able to filter on channel values. This string structure is not specified by the IPTC. ▪ **timestamp** (page 195) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.99 Narrower

Table 106. Narrower

(XML) Data Model	Element
Namespace (prefix)	nar
Name	narrower
Title	Narrower
Definition	An identifier of a more specific concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**RelatedConceptType** (page 237)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.100 News Coverage

Table 107. News Coverage

(XML) Data Model	Element			
Namespace (prefix)	nar			
Name	newsCoverage			
Title	News Coverage			
Definition	Textual information about the intended coverage by the news provider of this event information. This information is aimed at the editorial staff of the receiver.			
User Note(s)				
Implementation Note(s)				
XML Schema Spec	At: PCL			
Datatype				
Internally Ctrl Values				
Externally Ctrl Values				
Attribute(s)	▪ role (0..1); **QCodeType** (page 234); Refines the semantics of the property.			
	▪ **editAttributes** (page 243)	Name	Datatype	
		id (0..1)	XML Schema ID	
		creator (0..1)	QCodeType	
		modified (0..1)	DateOptTimeType	
Child Element(s)	▪ **edNote** (page 101) (0..unbounded)			
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.			
XML Schema Note(s)				
Example(s)				

11.6.101 News Item (NewsML-G2 Specific)

Table 108. News Item (NewsML-G2 Specific)

(XML) Data Model	Element
Namespace (prefix)	nar
Name	newsItem
Title	News Item (NewsML-G2 Specific)
Definition	An Item containing news-related information.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**AnyItemType** (page 205)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta** (page 76) (0..1) ▪ **partMeta** (page 162) (0..1) ▪ **assert** (page 55) (0..unbounded) ▪ **inlineRef** (page 128) (0..unbounded) ▪ **contentSet** (page 79) (0..1)
XML Schema Note(s)	
Example(s)	

11.6.102 News Message

Table 109. News Message

(XML) Data Model	Element
Namespace (prefix)	nar
Name	newsMessage
Title	News Message
Definition	A container to exchange one or more items.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **header** (page 148) (1)
	▪ **itemSet** (page 137) (1)
XML Schema Note(s)	
Example(s)	

11.6.103 Note

Table 110. Note

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	note		
Title	Note		
Definition	Additional natural-language information about the concept.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	**BlockType** (page 208)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 244)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.6.104 Occurrence Status

Table 111. Occurrence Status

(XML) Data Model	Element
Namespace (prefix)	nar
Name	occurStatus
Title	Occurrence Status
Definition	Indicates the certainty of the occurrence of the event.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QCodePropType** (page 233)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes: http://cv.iptc.org/newscodes/eventoccurstatus/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.105 Opening Hours

Table 112. Opening Hours

(XML) Data Model	Element
Namespace (prefix)	nar
Name	openHours
Title	Opening Hours
Definition	Opening-hours of the place, in natural language.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Label1Type** (page 229)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.106 Origin

Table 113. Origin

(XML) Data Model	Element
Namespace (prefix)	nar
Name	origin
Title	Origin
Definition	The point of origin of the transmission of the message.
User Note(s)	This string's structure is not specified by the IPTC.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.107 Organisation Details

Table 114. Organisation Details

(XML) Data Model	Element
Namespace (prefix)	nar
Name	organisationDetails
Title	Organisation Details
Definition	A group of properties specific to an organisation.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **founded** (page 94) (0..1) ▪ **dissolved** (page 93) (0..1) ▪ **location** (page 145) (0..unbounded) ▪ **contactInfo** (page 73) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.108 Organiser

Table 115. Organiser

(XML) Data Model	Element
Namespace (prefix)	nar
Name	organiser
Title	Organiser
Definition	A person or organisation organising the event.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PartyPropType** (page 217)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for the *role* attribute: http://cv.iptc.org/newscodes/eventorganiserrole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.109 Package Item

Table 116. Package Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	packageItem
Title	Package Item
Definition	An Item used for packaging references to other Items and Web resources.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**AnyItemType** (page 205)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta** (page 77) (0..1)
	▪ **assert** (page 55) (0..unbounded)
	▪ **inlineRef** (page 128) (0..unbounded)
	▪ **groupSet** (page 121) (0..1)
XML Schema Note(s)	
Example(s)	

11.6.110 Participant

Table 117. Participant

(XML) Data Model	Element
Namespace (prefix)	nar
Name	participant
Title	Participant
Definition	A person or organisation (e.g. a group of artists) participating in the event.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PartyPropType** (page 217)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes for *role* attribute: http://cv.iptc.org/newscodes/eventparticipantrole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.111 Participation Requirement

Table 118. Participation Requirement

(XML) Data Model	Element
Namespace (prefix)	nar
Name	participationRequirement
Title	Participation Requirement
Definition	A requirement for participating in the event.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeType** (page 234); Refines the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.112 Part Meta

Table 119. Part Meta

(XML) Data Model	Element
Namespace (prefix)	nar
Name	partMeta
Title	Part Meta
Definition	A set of properties associated with a specific part of the content of the Item.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ partid (0..1); XML Schema ID; The identifier of the part.

Attribute(s)	▪ seq (0..1); XML Schema nonNegativeInteger; The sequence number of the part.

		Name	**Datatype**
	▪ **i18nAttributes** (page 243)	xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	▪ **icon** (page 123) (0..1)		
	▪ **timeDelim** (page 194) (0..1)		
	▪ **regionDelim** (page 200) (0..1)		
		Element Name	**Page**
	▪ **Descriptive Metadata Group** (page 204) (0..1)	**by** (0..unbounded)	60
		dateline (0..unbounded)	96
		description (0..unbounded)	98
		genre (0..unbounded)	116
		headline (0..unbounded)	122
		language (0..unbounded)	140
		slugline (0..unbounded)	190
		subject (0..unbounded)	193
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		

XML Schema Note(s)	
Example(s)	

11.6.113 Person Details

Table 120. Person Details

(XML) Data Model	Element
Namespace (prefix)	nar
Name	personDetails
Title	Person Details
Definition	A group of properties specific to a person.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **born** (page 91) (0..1)
	▪ **died** (page 92) (0..1)
	▪ **affiliation** (page 51) (0..unbounded)
	▪ **contactInfo** (page 73) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.114 Phone Number

Table 121. Phone Number

(XML) Data Model	Element
Namespace (prefix)	nar
Name	phone
Title	Phone Number
Definition	An international phone number.
User Note(s)	The *tech* attribute indicates a land-line, cellular etc., service.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**ElectronicAddressTechType** (page 215)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.115 POI Details

Table 122. POI Details

(XML) Data Model	Element
Namespace (prefix)	nar
Name	POIDetails
Title	POI Details
Definition	A group of properties specific to a point of interest.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **position** (page 117) (0..1) ▪ **openHours** (page 155) (0..1) ▪ **capacity** (page 61) (0..1) ▪ **access** (page 47) (0..unbounded) ▪ **details** (page 146) (0..unbounded) ▪ **contactInfo** (page 73) (0..unbounded) ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.116 Postal Address

Table 123. Postal Address

(XML) Data Model	Element
Namespace (prefix)	nar
Name	address
Title	Postal Address
Definition	A postal address.
User Note(s)	A special value of the role attribute may indicate that this information is not used to make contacts but e.g. is the registered address of a company.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the postal address.
Child Element(s)	▪ **line** (page 50) (0..unbounded)
	▪ **locality** (page 143) (0..1)
	▪ **area** (page 84) (0..1)
	▪ **country** (page 83) (0..1)
	▪ **postalCode** (page 167) (0..1)
XML Schema Note(s)	
Example(s)	

11.6.117 Postal Code

Table 124. Postal Code

(XML) Data Model	Element
Namespace (prefix)	nar
Name	postalCode
Title	Postal Code
Definition	A postal code, part of the address.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**IntlStringType** (page 227)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.118 Priority

Table 125. Priority

(XML) Data Model	Element
Namespace (prefix)	nar
Name	priority
Title	Priority
Definition	The priority of this message in the overall transmission process. A value of 1 corresponds to the highest priority, a value of 9 to the lowest.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**Int1To9Type** (page 226)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.119 Profile

Table 126. Profile

(XML) Data Model	Element
Namespace (prefix)	nar
Name	profile
Title	Profile
Definition	The name of the structural template (aka profile) used for the generation of the Item.
User Note(s)	This property gives information about the precise structure of an Item, e.g. a simple package, article with one picture, and may be the name of the transformation stylesheet used for the generation of the Item.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**VersionedStringType** (page 242)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.120 Property Value Name

Table 127. Property Value Name

(XML) Data Model	Element
Namespace (prefix)	nar
Name	name
Title	Property Value Name
Definition	A natural-language name of the concept assigned as property value.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**IntlStringType** (page 227)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.121 Publish Status

Table 128. Publish Status

(XML) Data Model	Element
Namespace (prefix)	nar
Name	pubStatus
Title	Publish Status
Definition	The publishing status of the Item. If no value is provided the default value is "usable".
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QualPropType** (page 235)
Internally Ctrl Values	
Externally Ctrl Values	Mandatory IPTC NewsCodes: http://cv.iptc.org/newscodes/pubstatusg2/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.122 Recurrence Date

Table 129. Recurrence Date

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rDate
Title	Recurrence Date
Definition	An explicit date (and optionally time with the time zone) of recurrence.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**DateOptTimePropType** (page 211)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.123 Recurrence Rule

Table 130. Recurrence Rule

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rRule
Title	Recurrence Rule
Definition	A rule of recurrence.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**RecurrenceRuleType** (page 236)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.124 Registration

Table 131. Registration

(XML) Data Model	Element
Namespace (prefix)	nar
Name	registration
Title	Registration
Definition	How and when to register for the event. Could also include information about cost, and so on. May also hold accreditation information.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**BlockType** (page 208)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes: http://cv.iptc.org/newscodes/eventregrole/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.125 Related Concept

Table 132. Related Concept

(XML) Data Model	Element
Namespace (prefix)	nar
Name	related
Title	Related Concept
Definition	An identifier of a related concept, where the relationship is different from elements *sameAs*, *broader*, or *narrower*.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**RelatedConceptType** (page 237)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.126 Remote Catalog Reference

Table 133. Remote Catalog Reference

(XML) Data Model	Element
Namespace (prefix)	nar
Name	catalogRef
Title	Remote Catalog Reference
Definition	A reference to a remote catalog.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ href (1); **IRIType** (page 228); A hyperlink to a remote catalog.

Attribute(s)	▪ **editAttributes** (page 243)	Name	Datatype
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType

Child Element(s)	▪ **title** (page 138) (0..unbounded)
XML Schema Note(s)	
Example(s)	

11.6.127 Remote Content (NewsML-G2 Specific)

Table 134. Remote Content (NewsML-G2 Specific)

(XML) Data Model	Element
Namespace (prefix)	nar
Name	remoteContent
Title	Remote Content
Definition	A rendition of the content using a reference/link to a resource representing the content data at a remote location.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	

Table 134. Remote Content (NewsML-G2 Specific) (Continued)

	▪ href (1); **IRIType** (page 228); The reference of the asset representing the content.		
	▪ size (0..1); XML Schema nonNegativeInteger; The size in bytes of the remote content.		
	▪ language (0..1); XML Schema normalized string; The language used by the remote content.		
	▪ **newsContentAttributes** (page 244)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		rendition (0..1)	QCodeType
		contenttype (0..1)	XML Schema string
		format (0..1)	QCodeType
		generator (0..1)	XML Schema string
		generated (0..1)	DateOptTimeType
	▪ **timeValidityAttributes** (page 244)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Attribute(s)	▪ **newsContentCharacteristics** (page 245)	**Name**	**Datatype**
		wordcount (0..1)	XML Schema nonNegativeInteger
		width (0..1)	XML Schema nonNegativeInteger
		height (0..1)	XML Schema nonNegativeInteger
		orientation (0..1)	XML Schema nonNegativeInteger
		colourspace (0..1)	QCodeType
		resolution (0..1)	XML Schema positiveInteger
		duration (0..1)	XML Schema nonNegativeInteger
		audiocodec (0..1)	XML Schema nomalizedString
		audiobitrate (0..1)	XML Schema positiveInteger
		audiovbr (0..1)	enumeration: yes/no
		audiosamplesize (0..1)	XML Schema positiveInteger
		audiosamplerate (0..1)	XML Schema positiveInteger
		audiochannels (0..1)	QCodeType
		videocodec (0..1)	XML Schema nomalizedString
		videoavgbitrate (0..1)	XML Schema positiveInteger
		videovbr (0..1)	enumeration: yes/no
		videoframerate (0..1)	XML Schema positiveInteger

Table 134. Remote Content (NewsML-G2 Specific) (Continued)

Child Element(s)	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.6.128 Rights Information

Table 135. Rights Information

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	rightsInfo		
Title	Rights Information		
Definition	A set of properties representing the rights associated with the Item.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	idrefs; (0..1); XML Schema IDREFS; The reference to the part(s) of the content the rights are about.		
	• **timeValidityAttributes** (page 244)	Name	DataType
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)	▪ **accountable** (page 49) (0..1)		
	▪ **copyrightHolder** (page 81) (0..1)		
	▪ **copyrightNotice** (page 82) (0..unbounded)		
	▪ **usageTerms** (page 199) (0..unbounded)		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

11.6.129 Role in the Workflow

Table 136. Role in the Workflow

(XML) Data Model	Element
Namespace (prefix)	nar
Name	role
Title	Role in the Workflow
Definition	The role of the Item in the editorial workflow.
User Note(s)	Among other possibilities this property may indicate the importance of the item in a fee by concepts like "flash", "bulletin", "alert", "urgent", "newsbreak", and so on.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QualPropType** (page 235)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.130 Ruby

Table 137. Ruby

(XML) Data Model	Element			
Namespace (prefix)	nar			
Name	ruby			
Title	Ruby			
Definition	Ruby annotation for documents using an East Asian script.			
User Note(s)				
Implementation Note(s)	This implementation aligns with the Simple Ruby markup without and with parentheses of the W3C (see http://www.w3.org/TR/ruby/#simple-ruby1).			
XML Schema Spec	At: PCL			
Datatype				
Internally Ctrl Values				
Externally Ctrl Values				
Attribute(s)	• **editAttributes** (page 243)		Name	Datatype
			id (0..1)	XML Schema ID
			creator (0..1)	QCodeType
			modified (0..1)	DateOptTimeType
	• **i18nAttributes** (page 243)		**Name**	**Datatype**
			xml:lang (0..1)	XML Schema language
			dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	• **rb** (page 183) (1)			
	• **rp** (page 184) (see XML Schema note below)			
	• **rt** (page 185) (see XML Schema note below)			
XML Schema Note(s)	The alternative simple Ruby markup without and with parentheses is expressed by the use of either a single rt element or a single *rp-rt-rp* sequence of elements. Ruby parentheses (<rp>, empty elements) must be used as a pair: either both are present or none is present.			

Example(s)	`<ruby>`
	`   <rb>IPTC</rb>`
	`   <rp>(<rp/><rt>International Press Telecommunications`
	`   Council</rt><rp>)<rp/>`
	`</ruby>`			

11.6.131 Ruby Base

Table 138. Ruby Base

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rb
Title	Ruby Base
Definition	Ruby base text.
User Note(s)	Also see **ruby** (page 182).
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.132 Ruby Parenthesis

Table 139. Ruby Parenthesis

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rp
Title	Ruby Parenthesis
Definition	Visual parentheses for Ruby Text
User Note(s)	Also see **ruby** (page 182).
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	Ruby parentheses elements must be used as a pair: either both are present or none is present.

Example(s)	`<ruby>`
	`    <rb>IPTC</rb>`
	`    <rp>(<rp/><rt>International Press Telecommunications`
	`    Council</rt><rp>)<rp/>`
	`</ruby>`

11.6.133 Ruby Text

Table 140. Ruby Text

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rt
Title	Ruby Text
Definition	Ruby text.
User Note(s)	Also see **ruby** (page 182).
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.134 Same As

Table 141. Same As

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	sameAs		
Title	Same As		
Definition	An identifier of an equivalent concept.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	**TypedQualPropType** (page 241)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 244)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.6.135 Scheme Declaration

Table 142. Scheme Declaration

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	scheme		
Title	Scheme Declaration		
Definition	A scheme alias-to-URI mapping.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ alias (1); XML Schema NCName; A short string assigned by the provider as a representation of the scheme URI.		
	▪ uri (1); **IRIType** (page 228); The URI which identifies the scheme.		
	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.6.136 Sender

Table 143. Sender

(XML) Data Model	Element
Namespace (prefix)	nar
Name	sender
Title	Sender
Definition	The sender of the items, which may be an organisation or a person.
User Note(s)	The structure of this string is not specified by the IPTC. Best practice is to identify a sender by its domain name.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.137 Signal

Table 144. Signal

(XML) Data Model	Element
Namespace (prefix)	nar
Name	signal
Title	Signal
Definition	An instruction to the processor of this item that the content requires special handling.
User Note(s)	This property might indicate major rewriting of the content, important correction, urgent handling etc. The processor might be required to perform specific actions, depending on the contract between the provider and the recipient. Users should be alerted of the reception of an Item containing a signal by some UI mechanism (sound or display). An editorial note (edNote) may be used to convey additional natural language information related to the processing of the content.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**QualPropType** (page 235)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.138 Slugline

Table 145. Slugline

(XML) Data Model	Element
Namespace (prefix)	nar
Name	slugline
Title	Slugline
Definition	A sequence of tokens associated with the content. The interpretation is provider specific.
User Note(s)	*separator* providers may choose to use more complex separation rules. In such a case the meaning of the separators must be conveyed by some other means.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**IntlStringType** (page 227)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ separator (0..1); XML Schema string; The character string acting as a separator between the tokens in the slugline.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.139 Span

Table 146. Span

(XML) Data Model	Element
Namespace (prefix)	nar
Name	span
Title	Span
Definition	A generic mechanism for adding inline information to parts of the textual content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	

Attribute(s)	▪ class (0..1); NMTOKENS; List of classes.
	
	▪ **i18nAttributes** (page 243)
	    Name / **Datatype**
	    xml:lang (0..1) / XML Schema language
	    dir (0..1) / XML Schema string: enumeration *ltr*, *rtl*.
	
	▪ **editAttributes** (page 243)
	    Name / **Datatype**
	    id (0..1) / XML Schema ID
	    creator (0..1) / QCodeType
	    modified (0..1) / DateOptTimeType

Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.140 Start Date/Time

Table 147. Start Date/Time

(XML) Data Model	Element
Namespace (prefix)	nar
Name	start
Title	Start Date/Time
Definition	The date (and optionally the time with time zone) the event commences. This may be an exact or an approximative value.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**ApproximateDateTimePropType** (page 206)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.141 Subject

Table 148. Subject

(XML) Data Model	Element
Namespace (prefix)	nar
Name	subject
Title	Subject
Definition	An important topic of the content; what the content is about. For a Knowledge Item the content is the set of concepts, for an event the content is the event as such.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1ConceptPropType** (page 216)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.142 Time Delimiter

Table 149. Time Delimiter

(XML) Data Model	Element
Namespace (prefix)	nar
Name	timeDelim
Title	Time Delimiter
Definition	A delimiter for a piece of streaming media content by time expressions.
User Note(s)	The time unit may take the following values, taken from an IPTC defined controlled vocabulary: - timeCode: the format of the timestamps is hh:mm:ss:ff (ff for frames). - timeCodeDropFrame: the format of the timestamps is hh:mm:ss:ff (ff for frames). - editUnit: the format of the timestamps is a long unsigned integer. - normalPlayTime: the format of the timestamps is hh:mm:ss:mmm (mmm for milliseconds). If no time unit is present, the value editUnit MUST be assumed.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	Mandatory IPTC NewsCodes: http://cv.iptc.org/newscodes/timeunit/
Attribute(s)	▪ start (0..1); XML Schema string; The timestamp corresponding to the start of the part.

	▪ end (0..1); XML Schema string; The timestamp corresponding to the end of the part.

	▪ timeunit (0..1); **QCodeType** (page 234); The unit used for start and end timestamps.

Attribute(s)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType

Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.143 Timestamp

Table 150. Timestamp

(XML) Data Model	Element
Namespace (prefix)	nar
Name	timestamp
Title	Timestamp
Definition	A date-and-time associated with the message, other than the date-and-time the message was sent.
User Note(s)	The exact meaning may be refined by the role qualifier.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**DateTimePropType** (page 213)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeType** (page 234); A refinement of the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.144 Transmission Identifier

Table 151. Transmission Identifier

(XML) Data Model	Element
Namespace (prefix)	nar
Name	transmitId
Title	Transmission Identifier
Definition	The transmission identifier associated with the message.
User Note(s)	This string's structure is not specified by the IPTC. No two News Messages sent by the same sender on the same date may have the same identifier. In case of retransmission it is not required to update this identifier.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.145 Type of a Concept

Table 152. Type of a Concept

(XML) Data Model	Element
Namespace (prefix)	nar
Name	type
Title	Type of a Concept
Definition	The nature of the concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QualPropType** (page 235)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.146 Urgency

Table 153. Urgency

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	urgency		
Title	Urgency		
Definition	The editorial urgency of the content. A value of 1 corresponds to the highest urgency, a value of 9 to the lowest.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	**Int1To9Type** (page 226)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.6.147 Usage Terms

Table 154. Usage Terms

(XML) Data Model	Element
Namespace (prefix)	nar
Name	usageTerms
Title	Usage Terms
Definition	A natural-language statement about the usage terms pertaining to the content.
User Note(s)	This property includes the type of usage to which the rights apply, the geographical area or areas to which specified usage rights pertain, the indication of the rights holder, restrictions on the use of the content and the time period over which the stated rights apply. If no usage terms are specified, then no specific restrictions on use of the content beyond contractual ones are being asserted.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**RightsLabelType** (page 238)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.148 Visual Region Delimiter

Table 155. Visual Region Delimiter

(XML) Data Model	Element
Namespace (prefix)	nar
Name	regionDelim
Title	Visual Region Delimiter
Definition	A delimiter for a rectangular region in a piece of visual content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	A: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ x (0..1); XML Schema integer; The x-axis coordinate of the side of the rectangle which has the smaller x-axis coordinate value in the current user coordinate system. ▪ y (0..1); XML Schema integer; The y-axis coordinate of the side of the rectangle which has the smaller y-axis coordinate value in the current user coordinate system. ▪ width (0..1); XML Schema integer; The width of the rectangle. ▪ height (0..1); XML Schema integer; The height of the rectangle. <table><tr><td></td><td>**Name**</td><td>**Datatype**</td></tr><tr><td rowspan="3">▪ **editAttributes** (page 243)</td><td>id (0..1)</td><td>XML Schema ID</td></tr><tr><td>creator (0..1)</td><td>QCodeType</td></tr><tr><td>modified (0..1)</td><td>DateOptTimeType</td></tr></table>
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.6.149 Web Address

Table 156. Web Address

(XML) Data Model	Element
Namespace (prefix)	nar
Name	web
Title	Web Address
Definition	A Web address.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**IRIType** (page 228)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the web address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.7 Element Group Definitions

11.7.1 Recurrence Group

This group of properties defines the information required to specify a recurrence set. The recurrence set is the complete set of recurrence instances for a calendar component. The model follows the iCalendar specification [RFC2445].

At least one rDate or rRule element MUST be present. These elements MUST come first in the group. Then the exDate and exRule elements MAY be inserted in any order.

Table 157. Recurrence Group Elements

Element Title	Element Name	Card	Described on Page
Recurrence Date	**rDate**	(0..unbounded)	172
Recurrence Rule	**rRule**	(0..unbounded)	173
Exclusion Date	**exDate**	(0..unbounded)	110
Exclusion Rule	**exRule**	(0..unbounded)	111

11.7.2 Concept Definition Group

This group of properties defines a concept using free-text. The name property MUST come first, then the other elements may be inserted in any order.

Table 158. Concept Definition Group Elements

Element Title	Element Name	Card	Described on Page
Concept Name	**name**	(0..unbounded)	70
Concept Definition	**definition**	(0..unbounded)	67
Note	**note**	(0..unbounded)	153
Facet	**facet**	(0..unbounded)	112

11.7.3 Concept Relationships Group

This group of properties defines the relationship between a concept and other concepts. The elements may be inserted in any order.

Table 159. Concept Relationships Group Elements

Element Title	Element Name	Card	Described on Page
Same As	**sameAs**	(0..unbounded)	186
Broader	**broader**	(0..unbounded)	59
Narrower	**narrower**	(0..unbounded)	149
Related	**related**	(0..unbounded)	175

11.7.4 Entity Details Group

This group of aggregate components defines detailed properties for a specific type of concept. Only one element from this group MAY be present in the wrapping context.

Table 160. Entity Details Group Elements

Element Title	Element Name	Card	Described on Page
Person Details	**personDetails**	(1)	163
Organisation Details	**organisationDetails**	(1)	157
Geopolitical Area Details	**geoAreaDetails**	(1)	118
POI Details	**POIDetails**	(1)	165

11.7.5 Administrative Metadata Group

This group of properties is related to the administrative facet of content. The order of the elements in this group is flexible: The non-repeatable elements MUST come first, then the repeatable elements may be inserted in any order.

Table 161. Administrative Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Urgency	**urgency**	(0..1)	198
Date Content Created	**contentCreated**	(0..1)	86
Date Content Modified	**contentModified**	(0..1)	87
Located	**located**	(0..unbounded)	133
Information Source	**infoSource**	(0..unbounded)	198
Creator	**creator**	(0..unbounded)	85
Contributor	**contributor**	(0..unbounded)	80
Audience	**audience**	(0..unbounded)	56
Excluded Audience	**exclAudience**	(0..unbounded)	109
Alternative Identifier	**altId**	(0..unbounded)	52

11.7.6 Knowledge Descriptive Metadata Group

This group of properties is related to the descriptive facet of knowledge content. The order of the elements in this group is flexible: all elements are repeatable and may be inserted in any order.

Table 162. Knowledge Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Subject	**subject**	(0..unbounded)	193
Description	**description**	(0..unbounded)	98
Language	**language**	(0..unbounded)	140

11.7.7 Descriptive Metadata Group

This group of properties is related to the descriptive facet of news content. The order of the elements in this group is flexible: all elements are repeatable and may be inserted in any order.

Table 163. Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Language	**language**	(0..unbounded)	140
Genre	**genre**	(0..unbounded)	116
Subject	**subject**	(0..unbounded)	193
Slugline	**slugline**	(0..unbounded)	190
Headline	**headline**	(0..unbounded)	122
Dateline	**dateline**	(0..unbounded)	96
By	**by**	(0..unbounded)	60
Description	**description**	(0..unbounded)	98

11.7.8 Item Management Group

This group of properties is related to the management of Items. They MUST appear in the order of the table below.

Table 164. Item Management Group Elements

Element Title	Element Name	Card	Described on Page
Item Class	**itemClass**	(1)	134
Content Provider	**provider**	(1)	78
Date Item Version Created	**versionCreated**	(1)	90
Date Item First Created	**firstCreated**	(0..1)	89
Date Item Embargo Ends	**embargoed**	(0..1)	88
Publish Status	**pubStatus**	(0..1)	171
Role in the Workflow	**role**	(0..1)	181
File Name	**filename**	(0..1)	114
Generator Tool	**generator**	(0..1)	115
Profile	**profile**	(0..1)	169
Editorial Service	**service**	(0..unbounded)	102
Item Title	**title**	(0..unbounded)	138
Editorial Note	**edNote**	(0..unbounded)	101
Member Of	**memberOf**	(0..unbounded)	147
Instance Of	**instanceOf**	(0..unbounded)	131
Signal	**signal**	(0..unbounded)	189
Alternative Representation	**altRep**	(0..unbounded)	53

11.8 Datatype Definitions

11.8.1 Any Item Type

Table 165. Any Item Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	AnyItemType
Title	Any Item Type
Definition	An abstract class. All G2 items are inherited from this class.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ standard; (0..1); string value: default = "XML Schema string"; The IPTC standard to which the Item is conformant.
	▪ standardversion; (1); XML Schema string; restricted to the format "integer.integer"; The major-minor version of the XML schema specifying the Item.
	▪ conformance; (1); string value: fixed = "XML Schema string" - default = "core"; The conformance level to which the Item is conformant.
	▪ guid; (1); XML Schema string; The persistent, universally unique identifier for the Item.
	▪ version; (0..1); XML Schema positiveInteger; The version of the Item.

		Name	Datatype
	▪ i18nAttributes (page 243)	xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	▪ **catalogRef** (page 176) (0..unbounded)
	▪ **catalog** (page 62) (0..unbounded)
	▪ **rightsInfo** (page 180) (0..unbounded)
	▪ **itemMeta** (page 135) (1)
XML Schema Note(s)	At least one of the elements catalogRef or catalog element MUST be present. These elements MAY be inserted in any order.
Example(s)	

11.8.2 Approximate Date and Time Property Type

Table 166. Approximate Date and Time Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	ApproximateDateTimePropType
Title	Approximate Date and Time Property Type
Definition	A calendar date with an optional time (with time zone) part and an optional approximation range for the date.
User Note(s)	If a start and/or end attribute exists, then the date is approximate, else it is defined precisely by the property's date. If only the approximation start date is provided the range ends with the property value; if only the approximation end date is provided the approximation range starts with the property value.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Union of a XML Schema dateTime, date, gYearMonth, gYear, gMonth, gMonthDay, and gDay, with the addition of the following qualifiers.
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	• approxstart (0..1); **TruncatedDateTimeType** (page 240); The date (and optionally time) at which the approximation range begins. • approxend (0..1); **TruncatedDateTimeType** (page 240); The date (and optionally time) at which the approximation range ends. • **editAttributes** (page 243) <table><tr><th>Name</th><th>Datatype</th></tr><tr><td>id (0..1)</td><td>XML Schema ID</td></tr><tr><td>creator (0..1)</td><td>QCodeType</td></tr><tr><td>modified (0..1)</td><td>DateOptTimeType</td></tr></table>
Child Element(s)	
XML Schema Note(s)	
Example(s)	Examples of the format {reference date, range start date, range end date}: {2006-09-20, 2006-09-18, 2006-09-30} = on about 20 September 2006, not before the 18th, not after the 30th. {1855, 1850, 1860} = in about 1855, not before the 1850, not after the 1860. {--05-03, 1950, 1953} = on a 3 May, between 1950 and 1953.

11.8.3 Audience Type

Table 167. Audience Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	AudienceType
Title	Audience Type
Definition	An audience for the content.
User Note(s)	*significance*: 1 – corresponds to the highest significance. *significance*: 9 – corresponds to the lowest significance.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Extends **Flex1PropType** (page 218)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ significance (0..1); **Int1To9Type** (page 226). A qualifier which indicates the expected significance of the content for this specific audience. ▪ **quantifyAttributes** (page 243)
Child Element(s)	
XML Schema Note(s)	
Example(s)	

Attribute table within quantifyAttributes:

Name	Datatype
confidence (0..1)	Int100Type
relevance (0..1)	Int100Type
why (0..1)	QCodeType

11.8.4 Block Type

Table 168. Block Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	BlockType
Title	Block Type
Definition	Information about the content as natural language string with minimal markup and line breaks.
User Note(s)	Blocks are primarily used for notes, comments or instructions created by a news provider for use by recipient editorial teams.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML mixed content
Internally Ctrl Values	
Externally Ctrl Values	

Attribute(s)			
	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the block.		
	▪ media (0..1); XML Schema NMTOKENS; An indication of the target media type(s) values as defined by the Cascading Style Sheets (CSS) specification.		
		Name	**Datatype**
	▪ **editAttributes** (page 243)	id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
		Name	**Datatype**
	▪ **i18nAttributes** (page 243)	xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	
	▪ **a** (page 54) (0..unbounded)
	▪ **span** (page 191) (0..unbounded)
	▪ **ruby** (page 182) (0..unbounded)
	▪ **br** (page 141) (0..unbounded)
	▪ **inline** (page 124) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.

XML Schema Note(s)	
Example(s)	

11.8.5 Concept Identifier Type

Table 169. Concept Identifier Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	ConceptIdType
Title	Concept Identifier Type
Definition	The preferred unambiguous identifier for the concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ qcode (1); **QCodeType** (page 234); A qualified code which identifies a concept. ▪ created (0..1); **DateOptTimeType** (page 212); The date (and optionally the time) when the identifier was created. ▪ retired (0..1); **DateOptTimeType** (page 212); The date (and optionally the time) after which the concept identifier should no longer be applied as the value of a property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.6 Content Metadata Type

Table 170. Content Metadata Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	ContentMetadataType		
Title	Content Metadata Type		
Definition			
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 243)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **Administrative Metadata Group** (page 203) (0..1)	**Element Name**	**Page**
		urgency (0..1)	198
		contentCreated (0..1)	86
		contentModified (0..1)	87
		located (0..unbounded)	144
		infoSource (0..unbounded)	133
		creator (0..unbounded)	85
		contributor (0..unbounded)	80
		audience (0..unbounded)	56
		exclAudience (0..unbounded)	109
		altId (0..unbounded)	52
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

11.8.7 Date and Optional Time Property Type

Table 171. Date and Optional Time Property Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	DateOptTimePropType		
Title	Date and Optional Time Property Type		
Definition	A date plus optionally a time and a time zone.		
User Note(s)	The time may be expressed in Universal Time Coordinates (UTC), or in local time together with a time zone offset in hours and minutes.		
Implementation Note(s)	DateOptTimePropType is used as a property datatype.		
XML Schema Spec	At: PCL		
Datatype	The union of a XML schema dateTime and date.		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.8.8 Date and Optional Time Type

Table 172. Date and Optional Time Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	DateOptTimeType
Title	Date and Optional Time Type
Definition	A date plus optionally a time and a time zone.
User Note(s)	The time may be expressed in Universal Time Coordinates (UTC), or in local time together with a time zone offset in hours and minutes.
Implementation Note(s)	DateOptTimeType is used as a datatype for attributes only.
XML Schema Spec	At: Both CCL and PCL
Datatype	The union of a XML schema dateTime (year, month, day, hour, minute, second, optional decimal fraction of a second) and date (year, month and day plus an optional time zone indicator).
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.9 Date and Time Property Type

Table 173. Date and Time Property Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	DateTimePropType		
Title	Date and Time Property Type		
Definition	A date plus a mandatory time and an optional time zone.		
User Note(s)			
Implementation Note(s)	DateTimePropType is used as a property datatype.		
XML Schema Spec	At: PCL		
Datatype	XML Schema dateTime		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.8.10 Electronic Address Type

Table 174. Electronic Address Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	ElectronicAddressType
Title	Electronic Address Type
Definition	
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	• role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the electronic address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.11 Electronic Address Tech Type

Table 175. Electronic Address Tech Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	ElectronicAddressTechType
Title	Electronic Address Tech Type
Definition	
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the electronic address.
	▪ tech (0..1); **QCodeType** (page 234); The technical variant of the electronic address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.12 Flexible 1 Concept Property Type

Table 176. Flexible 1 Concept Property Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	Flex1ConceptPropType		
Title	Flexible 1 Concept Property Type		
Definition			
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	Extends **Flex1PropType** (page 218)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **quantifyAttributes** (page 243)	**Name**	**Datatype**
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
		why (0..1)	QCodeType
Child Element(s)	▪ **bag** (page 57) (0..1)		
XML Schema Note(s)			
Example(s)			

11.8.13 Flexible 1 Party Property Type

Table 177. Flexible 1 Party Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Flex1PartyPropType
Title	Flexible 1 Party Property Type
Definition	
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Extends **FlexPartyPropType** (page 222)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.14 Flexible 1 Property Type

Table 178. Flexible 1 Property Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	Flex1PropType		
Title	Flexible 1 Property Type		
Definition	Flexible generic data type for both controlled and uncontrolled values.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	• qcode (0..1); **QCodeType** (page 234); A qualified code assigned as a property value. or		
	• literal (0..1); XML Schema normalizedString; A free-text value assigned as a property value.		
	• type (0..1); **QCodeType** (page 234); The type of the concept assigned as controlled or uncontrolled property value.		
	• **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	• **i18nAttributes** (page 243)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	• **Concept Definition Group** (page 202) (0..1)	**Element Name**	**Page**
		name (1..unbounded)	70
		definition (0..unbounded)	67
		facet (0..unbounded)	112
		note (0..unbounded)	153
	• **Concept Relationships Group** (page 202) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	59
		narrower (0..unbounded)	149
		related (0..unbounded)	175
		sameAs (0..unbounded)	186
	• Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

11.8.15 Flexible Location Property Type

Table 179. Flexible Location Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	FlexLocationPropType
Title	Flexible Location Property Type
Definition	Flexible location (i.e. geo area or POI) data type for both controlled and uncontrolled values.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	

Attribute(s)			
	▪ qcode (0..1); **QCodeType** (page 234); A qualified code assigned as property value. or		
	▪ literal (0..1); XML Schema normalizedString; A free-text value assigned as property value.		
	▪ type (0..1); **QCodeType** (page 234); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 243)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)			
	▪ **Concept Definition Group** (page 202) (0..1)	**Element Name**	**Page**
		name (1..unbounded)	70
		definition (0..unbounded)	67
		facet (0..unbounded)	112
		note (0..unbounded)	153
	▪ **Concept Relationships Group** (page 202) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	59
		narrower (0..unbounded)	149
		related (0..unbounded)	175
		sameAs (0..unbounded)	186
	▪ **geoAreaDetails** (page 118) (0..1) or ▪ **POIDetails** (page 165) (0..1)		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		

Table 179. Flexible Location Property Type (Continued)

XML Schema Note(s)	
Example(s)	

11.8.16 Flexible Organisation Property Type

Table 180. Flexible Organisation Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	FlexOrganisationPropType
Title	Flexible Organisation Property Type
Definition	Flexible organisation data type for both controlled and uncontrolled values.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ qcode (0..1); **QCodeType** (page 234); A qualified code assigned as property value. or ▪ literal (0..1); XML Schema normalizedString; A free-text value assigned as property value. ▪ type (0..1); **QCodeType** (page 234); The type of the concept assigned as controlled or uncontrolled property value.

		Name	**Datatype**
	▪ **editAttributes** (page 243)	id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 243)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)

▪ **organisationDetails** (page 157) (0..1)

		Element Name	**Page**
	▪ **Concept Definition Group** (page 202) (0..1)	**name** (1..unbounded)	70
		definition (0..unbounded)	67
		facet (0..unbounded)	112
		note (0..unbounded)	153
		Element Name	**Page**
	▪ **Concept Relationships Group** (page 202) (0..1)	**broader** (0..unbounded)	59
		narrower (0..unbounded)	149
		related (0..unbounded)	175
		sameAs (0..unbounded)	186

▪ Extension Point (0..unbounded). Any set of provider-defined properties.

XML Schema Note(s)	
Example(s)	

11.8.17 Flexible Party Property Type

Table 181. Flexible Party Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	FlexPartyPropType
Title	Flexible Party Property Type
Definition	Flexible party (i.e. person or organisation) data type for both controlled and uncontrolled values.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	

Attribute(s)			
	▪ qcode (0..1); **QCodeType** (page 234); A qualified code assigned as property value. or ▪ literal (0..1); XML Schema normalizedString; A free-text value assigned as property value.		
	▪ type (0..1); **QCodeType** (page 234); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 243)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)			
	▪ **personDetails** (page 163) (0..1) or ▪ **organisationDetails** (page 157) (0..1)		
	▪ **Concept Definition Group** (page 202) (0..1)	**Element Name**	**Page**
		name (1..unbounded)	70
		definition (0..unbounded)	67
		facet (0..unbounded)	112
		note (0..unbounded)	153
	▪ **Concept Relationships Group** (page 202) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	59
		narrower (0..unbounded)	149
		related (0..unbounded)	175
		sameAs (0..unbounded)	186
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		

Table 181. Flexible Party Property Type (Continued)

XML Schema Note(s)	
Example(s)	

11.8.18 Flex Person Property Type

Table 182. Flex Person Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	FlexPersonPropType
Title	Flex Person Property Type
Definition	Flexible person data type for both controlled and uncontrolled values.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ qcode (0..1); **QCodeType** (page 234); A qualified code assigned as property value. or ▪ literal (0..1); XML Schema normalizedString; A free-text value assigned as property value. ▪ type (0..1); **QCodeType** (page 234); The type of the concept assigned as controlled or uncontrolled property value.
	editAttributes (page 243) — Name / Datatype: id (0..1) / XML Schema ID; creator (0..1) / QCodeType; modified (0..1) / DateOptTimeType
	i18nAttributes (page 243) — Name / Datatype: xml:lang (0..1) / XML Schema language; dir (0..1) / XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **personDetails** (page 163) (0..1) ▪ **Concept Definition Group** (page 202) (0..1) — Element Name / Page: **name** (1..unbounded) / 70; **definition** (0..unbounded) / 67; **facet** (0..unbounded) / 112; **note** (0..unbounded) / 153 ▪ **Concept Relationships Group** (page 202) (0..1) — Element Name / Page: **broader** (0..unbounded) / 59; **narrower** (0..unbounded) / 149; **related** (0..unbounded) / 175; **sameAs** (0..unbounded) / 186 ▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

11.8.19 Integer 0 to 100 Type

Table 183. Integer 0 to 100 Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Int100Type
Title	Integer 0 to 100 Type
Definition	An integer with a value range from 0 to 100.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema integer, value restriction 0 to 100.
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.20 Integer 1 to 9 Type

Table 184. Integer 1 to 9 Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Int1To9Type
Title	Integer 1 to 9 Type
Definition	An integer with a value range from 1 to 9.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema integer, value restriction 1 to 9.
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.21 International String Type

Table 185. International String Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	IntlStringType		
Title	International String Type		
Definition	An internationalized string, where the language and directionality in which the information is written are indicated.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 243)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.8.22 IRI Type

Table 186. IRI Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	IRIType
Title	IRI Type
Definition	An Internationalized Resource Identifier reference, as defined by RFC3987.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema anyURI
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.23 Label 1 Type

Table 187. Label 1 Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Label1Type
Title	Label 1 Type
Definition	Information about the content as natural language string with minimal markup.
User Note(s)	Labels are assertions expressed as natural language strings intended to be consumed by human beings. They are typically displayed alongside the content of an Item or in place of Items in a list, providing a means of selection among them.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML mixed content
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 232); A refinement of the semantics of the label.

Attribute(s) (cont.)	▪ media (0..1); XML Schema NMTOKENS; An indication of the target media type(s), values as defined by the Cascading Style Sheets (CSS) specification.		
		Name	**Datatype**
	editAttributes (page 243)	id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
		Name	**Datatype**
	i18nAttributes (page 243)	xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	▪ **a** (page 54) (0..unbounded)
	▪ **span** (page 191) (0..unbounded)
	▪ **ruby** (page 182) (0..unbounded)
	▪ **inline** (page 124) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	Anchor, Span and Ruby are modelled after their XHTML 1.1 counterparts.
Example(s)	

11.8.24 Link 1 Type

Table 188. Link 1 Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Link1Type
Title	Link 1 Type
Definition	
User Note(s)	If the target resource is an Item, the guidref attribute SHOULD identify the target Item by its globally unique identifier and a version attribute MAY indicate the target Item version; in the absence of version information, the target resource is the latest version available. The *href* attribute MAY additionally indicate the location of the target resource. If the target resource is a Web resource, the *href* attribute MUST indicate the locator of the target resource. The content type, a.k.a. IANA MIME type of the target resource MAY also be indicated by a *hreftype* attribute.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	see below
Child Element(s)	Extension Points (0..unbounded). Any set of hints, i.e. properties extracted from the target resource.

Attribute(s):

- guidref (0..1); XML Schema string; The globally unique Identifier of the target Item.
- href (0..1); **IRIType** (page 228); The locator of the target resource.
- hreftype (0..1); XML Schema normalizedString; Contains the IANA (Internet Assigned Numbers Authority) MIME type of the target of the link.
- version (0..1); XML Schema positiveInteger; The version of the target Item identified by a Target Item Identifier. By default, the latest revision is retrieved when the link is activated. Absent if the target is a Web resource.
- rel (0..1); **QCodeListType** (page 232); The identifier of the relationship between the current Item and the target resource.
- rank (0..1); XML Schema nonNegativeInteger; The rank of the link among other links.
- size (0..1); XML Schema nonNegativeInteger; The size in bytes of the target resource.

	Name	Datatype
editAttributes (page 243)	id (0..1)	XML Schema ID
	creator (0..1)	QCodeType
	modified (0..1)	DateOptTimeType

	Name	Datatype
i18nAttributes (page 243)	xml:lang (0..1)	XML Schema language
	dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

	Name	DataType
timeValidityAttributes (page 244)	validfrom (0..1)	DateOptTimeType
	validto (0..1)	DateOptTimeType

Table 188. Link 1 Type (Continued)

XML Schema Note(s)	Extension Point: a particular hint is a title, already defined at the CCL as a short natural language name representing the link and displayed to the users.
Example(s)	

11.8.25 QCode List Type

Table 189. QCode List Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	QCodeListType
Title	QCode List Type
Definition	A space-separated list of QCodeType values.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	List of QCodeType values.
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.26 QCode Property Type

Table 190. QCode Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	QCodePropType
Title	QCode Property Type
Definition	An element with a QCode value in a qcode attribute.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ qcode (1); **QCodeType** (page 234); A qualified code assigned as a property value. ▪ **editAttributes** (page 243) <table><tr><th>Name</th><th>Datatype</th></tr><tr><td>id (0..1)</td><td>XML Schema ID</td></tr><tr><td>creator (0..1)</td><td>QCodeType</td></tr><tr><td>modified (0..1)</td><td>DateOptTimeType</td></tr></table>
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.27 QCode Type

Table 191. QCode Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	QCodeType
Title	QCode Type
Definition	A QCode value.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	A set of characters (no whitespace, no colon) followed by a colon (:) character, followed by a set of characters with no whitespace. The corresponding regular expression is: [^\s:]+:[^\s]+
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.28 Qualified Property Type

Table 192. Qualified Property Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	QualPropType		
Title	Qualified Property Type		
Definition	An element with a QCode value and optional names.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	Extends **QCodePropType** (page 233)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 243)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **name** (page 70) (0..unbounded)		
XML Schema Note(s)			
Example(s)			

11.8.29 Recurrence Rule Type

Table 193. Recurrence Rule Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	RecurrenceRuleType
Title	Recurrence Rule Type
Definition	A rule of recurrence applied to a date associated with an event.
User Note(s)	The different datatypes listed in the Attribute(s) row below correspond to iCalendar datatypes and enumerations.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	• freq (1); XML Schema string; The type of recurrence rule.
	• interval (0..1); XML Schema positiveInteger; How often the recurrence rule repeats.
	• until (0..1); **DateOptTimeType** (page 212); A date-time value which bounds the recurrence rule in an inclusive manner.
	• count (0..1); XML Schema positiveInteger; The number of occurrences at which to range-bound the recurrence.
	• wkst (0..1); XML schema string, enumeration; The day on which the workweek starts.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.30 Related Concept Type

Table 194. Related Concept Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	RelatedConceptType
Title	Related Concept Type
Definition	An identifier of a related concept, where the relationship is different from elements *sameAs*, *broader*, or *narrower*.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Extends **TypedQualPropType** (page 241)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ rel (0..1); **QCodeType** (page 234); The identifier of the relationship between the current concept and the target concept.

		Name	**DataType**
Attribute(s)	▪ **timeValidityAttributes** (page 244)	validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType

Child Element(s)	**facet** (page 112) (0..unbounded)
XML Schema Note(s)	
Example(s)	

11.8.31 Rights Label Type

Table 195. Rights Label Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	RightsLabelType
Title	Rights Label Type
Definition	
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Label1Type** (page 229)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ href (0..1); **IRIType** (page 228); The locator of a remote expression of rights.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.32 Truncated Date and Time Property Type

Table 196. Truncated Date and Time Property Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	TruncatedDateTimePropType		
Title	Truncated Date and Time Property Type		
Definition	An element with a calendar date as a value. The date has an optional time part: it is optionally possible to omit one to many less significant components, from right to left. "From right to left" means starting from the least significant component (i.e. fraction of a second) and to continue with the full time part, the day part and the month part. The year part must not be omitted.		
User Note(s)			
Implementation Note(s)	TruncatedDateTimePropType is used as a property datatype.		
XML Schema Spec	At: PCL		
Datatype	The union of a XML Schema dateTime, date, gYearMonth and gYear, and additionally supports provider-defined qualifiers.		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 243)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

11.8.33 Truncated Date and Time Type

Table 197. Truncated Date and Time Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	TruncatedDateTimeType
Title	Truncated Date and Time Type
Definition	A calendar date with an optional time part: it is optionally possible to omit one to many less significant components, from right to left. "From right to left" means starting from the least significant component (i.e. fraction of a second) and to continue with the full time part, the day part and the month part. The year part must not be omitted.
User Note(s)	
Implementation Note(s)	TruncatedDateTimeType is used as a qualifier datatype.
XML Schema Spec	At: Both CCL and PCL
Datatype	The union of a XML Schema dateTime, date, gYearMonth and gYear.
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.34 Typed Qualified Property Type

Table 198. Typed Qualified Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	TypedQualPropType
Title	Typed Qualified Property Type
Definition	An element with a QCode value and an additional type for this value.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QualPropType** (page 235)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ type (0..1); **QCodeType** (page 234); The type of the concept assigned as property value.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.8.35 Versioned String Type

Table 199. Versioned String Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	VersionedStringType
Title	Versioned String Type
Definition	The type extending IntlStringType by a version information.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**IntlStringType** (page 227)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ versioninfo (0..1); XML schema string; The version of a processing resource.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

11.9 Attribute Group Definitions

11.9.1 Internationalization Attributes Group

Table 200. i18nAttributes

Title	Name	Card	Datatype	Definition
Language Indicator	xml:lang	0..1	XML Schema language	The language of textual content.
Direction	dir	0..1	XML Schema string: enumeration *ltr*, *rtl*.	The directionality of textual content.

Notes:

- ♦ *xml:lang* values MUST follow RFC 4646 and RFC 4647 (as both replace RFC 3066) or its successor. See also IETF BCP47.
- ♦ The *dir* qualifier specifies the directionality of scripted text: left-to-right ("ltr", the default) or right-to-left ("rtl"). Its definition follows the XHTML 1.0 production. Directionality – left-to-right or right-to-left – is assigned to characters in Unicode, in order to allow the text to be rendered properly. For example, while English characters are presented left-to-right, Hebrew characters are presented right-to-left. Unicode defines a bidirectional algorithm that must be applied whenever a document contains right-to-left characters. While this algorithm usually gives the proper presentation, some situations leave directionally neutral text and require the dir attribute to specify the base directionality.

11.9.2 Editing Attributes Group
These attributes are used only at PCL.

Table 201. editAttributes

Title	Name	Card	Datatype	Definition
Local Identifier	id	0..1	XML Schema ID	The local identifier of the element.
Creator	creator	0..1	QCodeType	If the supporting property has no value, specifies which party (person, organisation or system) will edit the property. If the supporting property has a value, specifies which party (person, organisation or system) has edited the property.
Date Modified	modified	0..1	DateOptTimeType	The date (and, optionally, the time) when the property was last modified. The initial value is the date (and, optionally, the time) of creation of the property.

11.9.3 Quantify Attributes Group
These attributes are used only at PCL.

Table 202. quantifyAttributes

Title	Name	Card	Datatype	Definition
Confidence	confidence	0..1	Int100Type	The confidence with which the metadata has been assigned.
Relevance	relevance	0..1	Int100Type	The relevance of the metadata to the news content to which it is attached.
Why	why	0..1	QCodeType	Why the metadata has been included.

Notes:

♦ An indication of confidence is usually obtains by automatic categorization means. 100 is the highest value.

♦ A high relevance indicates that this piece of metadata truly expresses what the piece of news is about, while a low relevance indicates a low correlation between the metadata and the essence of the piece of news.

♦ *why* indicates whether the metadata is directly extracted from the content by a tool and/or by a person, whether it is an ancestor of some other concept directly associated with the content (e.g. the concepts France and Europe are ancestors of the concept Paris), or whether it is derived by look-up in a thesaurus (e.g. the entity Merck may be associated with the concept Pharmaceutical Industry Sector).

11.9.4 Time Validity Attributes Group

These attributes are used only at PCL.

Table 203. timeValidityAttributes

Title	Name	Card	Datatype	Definition
Valid From	validfrom	0..1	DateOptTimeType	The date (and optionally the time) *before* which a relationship is not valid.
Valid To	validto	0..1	DateOptTimeType	The date (and optionally the time) *after* which a relationship is not valid.

11.9.5 News Content Attributes

Table 204. newsContentAttributes

Title	Name	Card	Datatype	Definition
Local Identifier	id	0..1	XML Schema ID	The local identifier of the element.
Rendition	rendition	0..1	QCodeType	The specific rendition of content this component represents.
Content Type	contenttype	0..1	XML Schema string	An IANA MIME type.
Format	format	0..1	QCodeType	A refinement of a generic Content Type.
Generator tool	generator	0..1	XML Schema string	The name and version of the software tool used to generate the content.
Generated	generated	0..1	DateOptTimeType	The date (and, optionally, the time) when the content was generated.

Notes:

♦ *rendition* helps the processor choosing between alternative content components. Thus a picture may have pieces of content rendered as "thumbnail" or "preview", a text Item may contain an "sms", a "web" and a "print" rendition; values may be extended by individual providers.

♦ *contenttype* applies to the content before any technical encoding needed to make the data XML-compliant (e.g. base64).

♦ Note that *contentype* and *itemClass* of **Item Metadata** (page 135) are complementary. *itemClass* indicates the nature of the Item's content, but not the format of the components it contains: an Item can be of class "video" with a gif thumbnail and a mpeg2 main rendition.

♦ format is used if no precise content type exists exists (e.g. "application/xml" or "text/plain" are the only mime types available for a given format). In such a case the Content Type information is complement with Format information. For example the NSK variant of the TIFF format can be expressed as: Content Type = "image/tiff" plus Format = "fmt:NSk".

11.9.6 News Content Characteristics

To be implemented as an attribute group.

Table 205. newsContentCharacteristics

Title	Name	Card	Datatype	Definition
Word Count	wordcount	0..1	XML Schema nonNegativeInteger	The count of words of textual content. Applies to textual content.
Image Width	width	0..1	XML Schema nonNegativeInteger	The width of an image in pixels. Applies to image content. For video content, this is the number of pixels per lines.
Image Height	height	0..1	XML Schema nonNegativeInteger	The height of an image in pixels. Applies to image content. For video content, this is the number of lines per frame.
Image Orientation	orientation	0..1	XML Schema nonNegativeInteger	The orientation of the visual content of an image in regard to the standard rendition of the digital image data. Values in the range of 1 to 8 are compatible with the TIFF 6.0 and Exif 2.3 specifications. Applies to image content. Details about the values can be found in **Table 206**.
Image Colour Space	colourspace	0..1	QCodeType	The colour space of an image. Applies to image content.
Resolution	resolution	0..1	XML Schema positiveInteger	The recommended printing resolution for an image in dots per inch. Applies to image content.
Duration	duration	0..1	XML Schema nonNegativeInteger	The clip duration in seconds. Applies to audio-visual content.
Audio Codec	audiocodec	0..1	QCodeType	The applicable codec for audio data. Applies to audio content.
Audio Bit Rate	audiobitrate	0..1	XML Schema positiveInteger	The audio bit rate in Kbps. Applies to audio content.
Audio Variable Bit Rate flag	audiovbr	0..1	XML Schema boolean	An indication that the audio data is encoded with a variable bit rate. Applies to audio content.
Audio Sample Size	audiosamplesize	0..1	XML Schema positiveInteger	The number of bits per audio sample, e.g. 16. Applies to audio content. Aliases: audio bits per sample, audio resolution, audio encoding depth.
Audio Sample Rate	audiosamplerate	0..1	XML Schema positiveInteger	The number of audio samples per second, expressed as a sampling frequency in Hz, e.g. 44100. Applies to audio content.
Audio Channels	audiochannels	0..1	QCodeType	The audio sound system, e.g. *mono*, *stereo*, *surround*. Codes may represent e.g. *mono*, *stereo*, *surround*. Applies to audio content.
Video Codec	videocodec	0..1	QCodeType	The applicable codec for video data. Applies to video content.
Video Average Bit Rate	videoavgbitrate	0..1	XML Schema positiveInteger	The video average bit rate in Kbps. Used when the bit rate is variable. Applies to video content.
Video Variable Bit Rate flag	videovbr	0..1	XML Schema boolean	An indication that video data is encoded with a variable bit rate. Applies to video content.

Table 205. newsContentCharacteristics (Continued)

Title	Name	Card	Datatype	Definition
Video Frame Rate	videoframerate	0..1	XML Schema positiveInteger	The number of video frames per second, i.e. the rate at which the material should be shown in order to achieve the intended visual effect. This is the rate at which the material should be shown in order to achieve the intended visual effect. Applies to video content.
Video Scan Technique	videoscan	0..1	enumeration progressive/ interlaced	The video scan technique, progressive or interlaced. Applies to video content.
Video Aspect Ratio	videoaspectratio	0..1	XML Schema nomalizedString	The video aspect ratio, e.g. 4:3 or 16:9. Applies to video content.
Video Sampling Method	videosampling	0..1	XML Schema nomalizedString	The video sampling method, e.g. 4:1:1. Applies to video content.

Table 206 enumerates the allowed values for the **orientation** attribute. The values are integers from 1 to 8 and reflect the TIFF 6.0 and Exif 2.3 specification.

Remark on the Definition column: by the Exif specification the "0th row" is the first row which has been scanned for the digital image and the "0th column" the first column. The explanation describes how a picture having the orientation of value 1 has to be flipped and/or rotated to align with one of the values 2 through 8.

The column "Visual example" shows a picture of the character F having an orientation aligning with the value. For the viewer's convenience the visual top, bottom, left and right of an image are marked up with the initial character of the corresponding term.

Table 206. Orientation Values

Value	Definition and Explanation	Visual Example
1	The 0th row is at the visual top of the image, and the 0th column is the visual left-hand side. Explanation: upright, no flip, no rotation.	
2	The 0th row is at the visual top of the image, and the 0th column is the visual right-hand side. Explanation: upright, image flipped about the vertical axis.	
3	The 0th row is at the visual bottom of the image, and the 0th column is the visual right-hand side. Explanation: image rotated 180 degrees.	
4	The 0th row is at the visual bottom of the image, and the 0th column is the visual left-hand side. Explanation: image flipped about the vertical axis and rotated 180 degrees.	

Table 206. Orientation Values (Continued)

Value	Definition and Explanation	Visual Example
5	The 0th row is the visual left-hand side of the image, and the 0th column is the visual top. Explanation: image flipped about the vertical axis and rotated 90 degrees counterclockwise.	
6	The 0th row is the visual right-hand side of the image, and the 0th column is the visual top. Explanation: image rotated 90 degrees counterclockwise.	
7	The 0th row is the visual right-hand side of the image, and the 0th column is the visual bottom. Explanation: image flipped about the vertical axis and rotated 90 degrees clockwise.	
8	The 0th row is the visual left-hand side of the image, and the 0th column is the visual bottom. Explanation: image rotated 90 degrees clockwise.	

12 Glossary

Table 207. Glossary

Term	Definition
alias	See **scheme alias**.
anonymous controlled vocabulary	A **controlled vocabulary** that is not a **scheme**.
catalog	A file containing information about **scheme**(s).
code	A character sequence which forms a member of a **controlled vocabulary**.
concept	Anything that one may wish to refer to, e.g. Diplomacy, Paris, the Euro, OECD, the Japanese language, the IMF, Oil, Madonna, Olympic Games. Thus concept here has a broader meaning than is usual. This is because we are dealing with the idea of Paris, rather than with Paris itself, the idea of Oil, rather than Oil itself, and so on. Concepts fall in two broad categories: **named entity** and generic (or abstract) concepts. A concept may be defined by a **ConceptItem**.
ConceptItem	A specialised data structure containing data representing a **concept**. An identifier for the concept is mandatory and it may, optionally, provide information such as name, definition, relationships, etc. A **concept** defined by a ConceptItem is identified by a {**scheme alias**, **code**} pair. The reverse relationship does not necessarily hold. In other words, there is no requirement that each {scheme alias, code} pair has a corresponding ConceptItem. See also: **representation of a ConceptItem**.
concept type	A concept type allows the logical grouping of all similar **concept**(s), regardless of the **scheme**(s) the concepts belong to. Examples of concept type might be: Person, Organisation, Language, Business Sector, News Subject or Geography. A concept type is itself a concept and, as such, is represented by a code in a scheme.
concept URI	A URI which identifies a **concept**. A concept URI is obtained by appending the code representing this concept to the **scheme URI** corresponding to the scheme to which the code belongs. An abbreviated notation of a concept URI is a **Qualified code, QCode**.
conformance level	A layer of functionality defined by a standard. The **News Architecture** power conformance level is a superset of the **News Architecture** core conformance level, both in terms of structure and processing.
controlled vocabulary	A set of **code**(s), managed by some authority (e.g. a person or an organisation), employing some mechanism (e.g. an XML Schema, a Web page, an RFC, or **KnowledgeItem**) to maintain this set. A controlled vocabulary is either a **scheme** or is anonymous (i.e. an **anonymous controlled vocabulary**). Each code in a controlled vocabulary represents a **concept**.
constrained metadata container	A **metadata container** which either accepts only **code**(s) of a specified **concept type** or accepts only codes from a specified **controlled vocabulary** (which may be an **anonymous controlled vocabulary** or a **scheme**).
Definition	A human-readable string, held within a **ConceptItem**, which defines the **concept** which the item represents. Definitions will be implemented using **free-form text**.
formal metadata element	A metadata element designed to hold data that is not **free-form text**, e.g. **code**(s), or **formal text**. Such data is usually consumed by software. An example of such an element with a code value is subject. An example value of *subject* is "nc:15062000".

Table 207. Glossary (Continued)

Term	Definition
free-form metadata element	A metadata element designed to hold **free-form text**. Such data is usually consumed by humans. An example of a free-form metadata element is title. An example value of title is "Ian Thorpe makes a splash". The **News Architecture** provides a couple of datatypes for free-form text, e.g. International String, Label or BlockText.
free-form text	Arbitrary text, i.e. text which does not consist of **code**(s) drawn from a **controlled vocabulary**. A headline or a description is an example of free-form text.
formal text	A set of one or more **metadata container**(s) for **free-form text** to express formal information about a specific **concept**, but without identifying it. Basic properties for formal text are literal, name, definition and note. An example for formal text is the Creator property with a value of name = "Alfred Hitchcock", definition = "Suspense movie director and producer, born 1899, died 1980".
globally unique identifier	An identifier that is unique, unambiguous, and persistent. Being unique and unambiguous means that there is a 1:1 relationship between the identifier and the identified object. Being persistent means that the identifier never changes as time passes, and that it is never reused as an identifier for another object even if the original object disappears. See also **persistent identifier**, **unambiguous identifier**, and **unique identifier**.
Identifier	A string used to identify a specific **resource**. See **persistent identifier**, **unambiguous identifier**, **unique identifier**, and **globally unique identifier** (GUID).
KnowledgeItem	A Knowledge Item is a set of **concept** definitions to form a consistent structure, which is managed, protected and published as a whole. It facilitates the management and exchange of **controlled vocabulary**(ies).
Label	A generic term for datatypes designed to hold **free-form text**.
Metadata	Data which asserts something about some other data.
metadata container	A location (e.g. an element or an attribute) in a data structure, designed to hold **Metadata**. In XML it may be implemented as a **metadata element**.
metadata element	An XML element, which is either a **formal metadata element** or a **free-form metadata element**, it implements the notion of a **metadata container**.
named entity	A named entity may be a person, place, event, organization, product name, object name or any other news-related real life entity.
News Architecture	A framework of specifications common to all IPTC news exchange standards of the G2 Family of Standards.
news provider	A provider of news content, the entity responsible for the management of news items May be a news agency, a syndication company, a newspaper, a magazine ... or a blogger.
ontology	See **taxonomy**.
persistent identifier	An identifier which is associated with the same resource for all time. See also **unambiguous identifier**, **unique identifier**, and **globally unique identifier** (GUID).
processor	An application that supports the handling and processing of Items. Also known as a user agent.
property	A synonym term for a **metadata container** – may be implemented as XML element.
provider	See **news provider**.
publish	Make available to other parties involved in the news exchange process, according to the business practices of the provider.

Table 207. Glossary (Continued)

Term	Definition
Qualified code, QCode	A **concept URI** represented by a string of the form sss:ccc, where sss is a **scheme alias** and ccc is a **code**. Examples are iso4217:USD, rfc3066:zh-Hant, nc:15062000, nasdaq:msft and cusip:594918104. A QCode is not the same as a QName (qualified name) [W3C: Namespaces in XML (http:/ /www.w3.org/TR/REC-xml-names/)], though there are substantial similarities. The two main differences are: (i) the code does not have to be a valid XML name (e.g. can start with a digit), and (ii) the scheme alias does not have to be declared using a namespace declaration.
representation	The physical form of something.
representation of a ConceptItem	A manifestation of a given **ConceptItem** that is suited for some particular purpose. The various representations of a given ConceptItem may differ, for example, in whether they are verbose or concise, or in which language(s) they use for name and definition.
resource	A resource is a set of data that has identity.
scheme	A **controlled vocabulary** which is identified by a scheme URI. A scheme is not an anonymous controlled vocabulary.
scheme alias	A character sequence which is used as an abbreviation for a **scheme URI**. A scheme alias is similar but not identical to an XML Namespace prefix.
scheme URI	The URI which identifies the **scheme**. It is recommended to make this URI a URL and resolving it should result in retrieving information about the scheme.
synonym	Synonyms are **concept URI**(s) that refer from one **concept** to another concept with equivalent semantics. Synonymy is a symmetric relationship, which means that if A is synonymous with B, then B is also synonymous with A. An example of synonyms is "cemetery" and "graveyard". In the **News Architecture** synonyms are expressed by the **sameAs** (page 186) property.
target	The data being described by the metadata. The IPTC has chosen to use the term target rather than subject (the term used by RDF [http://www.w3.org/RDF/]), as subject has a special meaning in the context of News.
taxonomy	In a broad sense, taxonomy is the science of classification, but is often taken to mean a particular classification. In the context of the **News Architecture**, a taxonomy is a collection of **concept**(s), with associated **code**(s). A taxonomy may support typed relationships between concepts. Such a taxonomy is sometimes known as an **ontology** or **thesaurus**.
thesaurus	See **taxonomy**.
tuple	A set of values. The word tuple is a generalisation of the sequence: couple, triple, quadruple, quintuple, sextuple, etc. Tuples are conventionally written as a comma-separated list of items, enclosed within braces, e.g. {**scheme alias**, **code**}.
type	See **concept type**.
unambiguous identifier	An identifier is unambiguous if it identifies one and only one object (but an object may have several different unambiguous identifiers). See also **globally unique identifier**.
unconstrained metadata container	A **metadata container** that accepts **code**(s) from any **controlled vocabulary** and of any **concept type**.
unique identifier	The only identifier of a resource. See also **persistent identifier**, **unambiguous identifier**, and **globally unique identifier** (GUID)
Web resource	The data content that can be retrieved from a Web server using a Web-compliant transport protocol. See also **resource**.

13 References

Table 208. References

Subject	Description
IPTC Documents	
NML-BR	IPTC NewsML 2 Business Requirements: http://www.newsml.org/dl.php?fn=NewsML/2.0/specification/NewsML_2.0-spec-BusinessRequirements_1.pdf
EventsML-G2	Specifications for EventsML-G2: http://www.iptc.org/std/EventsML- G2/1.0/specification/
NewsML-G2	Specifications for NewsML-G2: http://www.iptc.org/std/NewsML-G2/2.0/specification
IPTC NewsCodes	All IPTC codes to categorise content or to express functional features can be obtained as NewsCodes from: http://www.newscodes.org
Other References	
XMLSCHEMA-1.0	W3C XML Schema 1.0 specifications at: http://www.w3.org/TR/xmlschema-1/, http://www.w3.org/TR/xmlschema-2/
XSD	W3C XML schema: http://www.w3.org/XML/Schema
XMLDSIG	XML-Signature Syntax and Processing: http://www.w3.org/TR/xmldsig-core/
RDF	Resource Description Framework (RDF): http://www.w3.org/RDF/
BCP47	Tags for Identifying Languages, IETF: http://rfc.net/bcp47.html
iCalendar	iCalendar as specified by RFC 2445: http://www.ietf.org/rfc/rfc2445.txt

14 Index

14.1 Elements

14.2 Datatypes

Contact Information

Contact the IPTC by:

Postal mail:
20 Garrick Street
London WC2E 9BT
United Kingdom

Web:

www.iptc.org

Email:

office@iptc.org