IPTC Standards

NewsML™ G2

EventsML G2

Specification Version 2.9

Power Conformance Level

Document Revision 1

Copyrights and License

Acknowledgements

This documentation is the result of a team effort by members of the International Press Telecommunications Council, with input and assistance from other contributors.

The effort to develop NewsML-G2 was lead by Laurent Le Meur (Agence France-Presse) and these persons contributed (ordered by family name): Yannick Beynet (Agence France-Presse), Mark Birbeck (x-port.net Ltd.), Dave Compton (Reuters), Jay Cousins (RivCom), Jean-Pierre Evain (EBU), John Evans (Transtel), Takahiro Fujiwara (EAST Co. Ltd.), Andreas Gebhard (Getty Images), Darko Gulija (HINA), Paul Harman (Press Association), Johan Lindgren (Tidningarnas Telegrambyrå), Jayson Lorenzen (BusinessWire), Stuart Myles (initally Dow Jones, now Associated Press), Kalle Rathje (dpa), Robert Schmidt-Nia (dpa), Michael Steidl (IPTC Managing Director), Ulf Wingstedt (CNET), Misha Wolf (Reuters) .

The Technical Writer of the initial version of the document was Scott Meltzer, now it is maintained by Michael Steidl.

About the Standards

Specification Versioning History

Version	Issue Date	Approved by	Remarks
2.0	2008-01-31	IPTC Standards Committee	NewsML-G2 approval
2.1	2008-07-03	IPTC Standards Committee	NewsML-G2 approval
2.4	2009-06-17	IPTC Standards Committee	NewsML-G2 approval
2.7	2010-06-30	IPTC Standards Committee	NewsML-G2 approval
1.6	2010-06-30	IPTC Standards Committee	EventsML-G2 approval
2.9	2011-06-09	IPTC Standards Committee	joint NewsML-G2/EventsML-G2 approval

The specifications of NewsML-G2 and EventsML-G2 have been published separetly up to the standard versions EvenstML-G2 1.7 and NewsML-G2 2.8. As the design and a vast majority of the specified structures are shared between both standards the IPTC decided in June 2011 to merge the specifications into a single document and to provide all specifications and other documentation only by the NewsML-G2 folders of the IPTC web server, see below Status of this Document.
This step has no impact on the structure of EventsML-G2 or NewsML-G2.

Document Revision History

Revision	Issue Date	Author (revised by)	Remark
1	2011-11-07	Michael Steidl	

About this Document

This document specifies the IPTC news exchange standard NewsML-G2 and its event focussed sibling 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 general news and event data.

Status of this Document

This document is under the governance of the IPTC News Exchange Formats Working Party and its sub-groups NewsML-G2 Working Group and EventsML-G2 Working Group.

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/NewsML-G2/2.9/

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

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

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

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

The Full Set of Specification Documents

The full set of specification documents for NewsML-G2 2.4 consists of (file names are added, # is to be substituted by the most current document revision number):

This Specification document - NewsML-G2_2.4-spec-PCL_#.pdf

XML Schema files applicable to the Core Conformance Level (see **Conformance Levels** on page 11):

- NewsML-G2_2.9-spec-All-Core_#.xsd

XML Schema files applicable to the Power Conformance Level (see **Conformance Levels** on page 11):

- NewsML-G2_2.9-spec-All-Power_#.xsd

All files above can be obtained from:

http://www.iptc.org/std/NewsML-G2/2.9/specification/

Note on the XML Schema File Names

XML Schemas are revised for two reasons:

- The NewsML-G2 specifications have been changed: this results in an new version of the standard, this will be reflected by a new path to files and a new standard version number like NewsML-G2_2.5
- 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 NewsML-G2_2.4-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 applying a persistent reference to the latest XML Schema file version regardless of any edits of the document.

Contents

1 Introduction to NewsML-G2

NewsML™ is a media-independent news exchange format for general news.

News exchange is a method of moving around not only the core news content, but also data that describe the content in an abstract way (i.e. metadata), information about how to handle news in an appropriate way (i.e.news management data), information about the packaging of news information, and finally information about the technical transfer itself.

1.1 History

The initial version of NewsML, version 1.0, was approved in October 2000. Since then it went along with minor revisions: version 1.1 was approved in October 2002; version 1.2 was approved in October 2003.

In 2004, the user-experience with NewsML was evaluated by the IPTC, and it was decided to create a consistent set of complementary standards as a comprehensive and interoperable way to move all types of data between media systems in order to make news exchange efficient and reliable. This set of standards is now the IPTC family of G2-Standards, and NewsML-G2 is a member of it.

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

To better understand the terminology used in the G2-Standards specifications we recommend the **Glossary** (page 315) as a reference, as it provides an extensive set of terms and their definitions.

The NewsML-G2 Specification builds on:

- The NewsML 2 Business Requirements [NML-BR],
- The IPTC-G2 News Architecture Core Model [NAR-CM],
- The IPTC-G2 News Architecture Power Extensions Model [NAR-PM].

1.2 Conformance Levels

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 implement 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.

2 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).

2.1 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.

2.2 Indication of Compliance with a Standard and Conformance Level

The IPTC **newsItem** (page 199) *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>
```

2.3 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>
```

2.4 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** (page 82) 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 (see **catalogRef** on page 232).

A remote catalog 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 URI 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 remote catalog elements MAY appear in any order, and their order is not relevant.

The main reason for using a sameAs indicator for a scheme in the catalog is speeding up QCode processing: a G2 processor does not have to check the individual concept for its sameAs relationships but can apply this relationship directly if the scheme identifier of the concept (used as property value) matches the scheme identifier with the sameAs child in the catalog.

Another reason for establishing a sameAs relationship between a scheme A of a provider and a referenced scheme B is to provide additional information about concepts, this could be identical information from scheme B in a different language or deeper information in the same language(s) as available with scheme B.

Detailed information on the structure of catalogs and their processing is given in **Dealing with Controlled Values** (page 50).

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>
```

2.5 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).

2.6 Rights Information

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

A **rightsInfo** (page 237) 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** (page 66) person, a **copyrightHolder** (page 105) and a set of **copyrightNotice** (page 106) elements and **usageTerms** (page 260).

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 a set of optional attributes (idrefs, scope, aspect), which indicate which part of the content is bound to these rights. Please review the comprehensive Processing Model below.

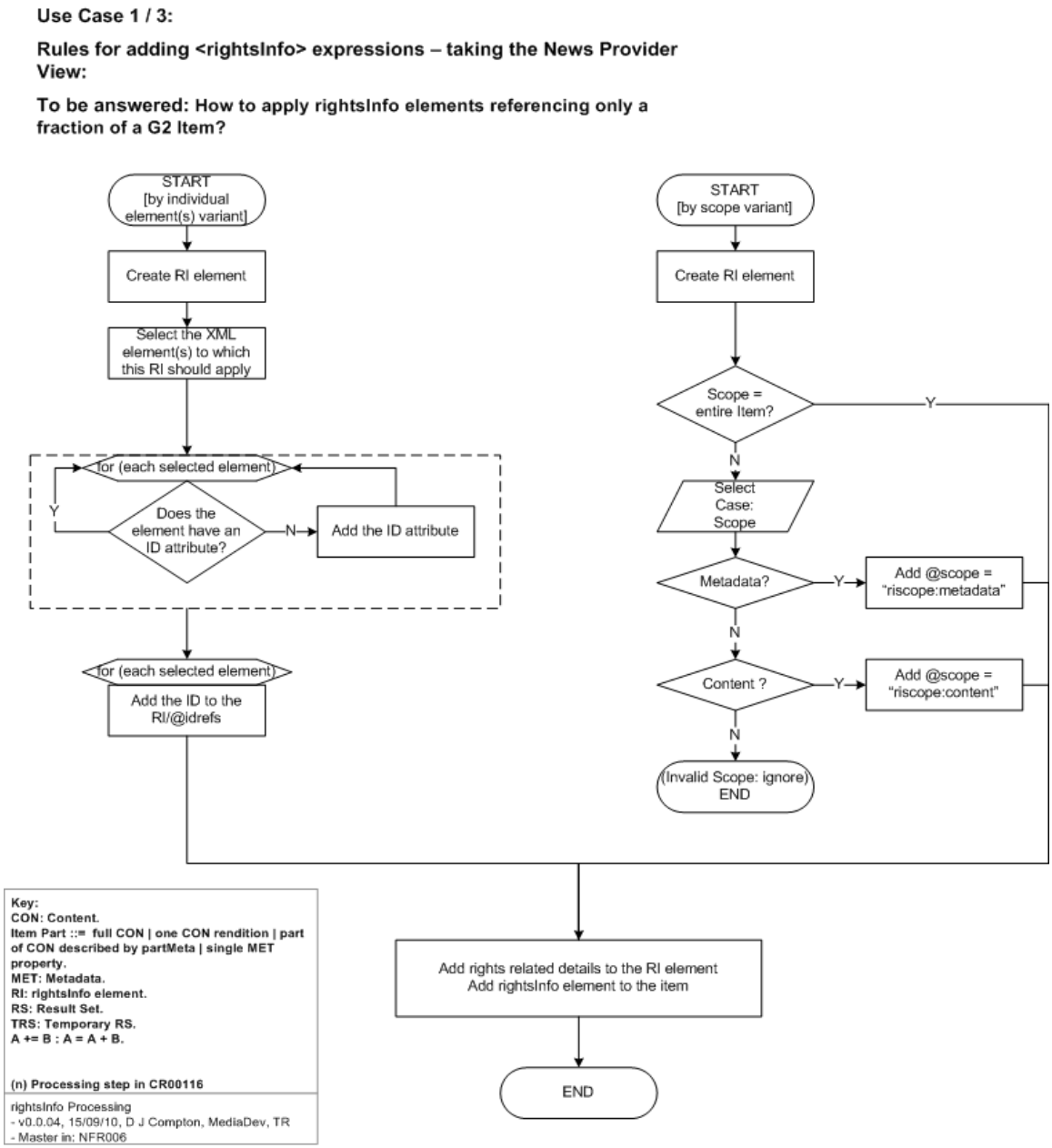
The rightsInfo element also provides optional time validity attributes (*validfrom* and *validto*) which express the date and time between which the set of rights properties apply.

Each provider may add a set of metadata properties which have to be defined in a non-IPTC-G2 namespace. See also **XML Namespaces** (page 60) and **Extension Points in XML** (page 61).

2.6.1 Processing Model

Rules for adding rightsInfo expressions - taking the News Provider View:

To be answered: Which markup does a provider has to apply?



1. How a rightsInfo element applies to an Item can be refined in two ways: a) making a statement about the scope, i.e. whether this rightsInfo element applies to the whole or part(s) of the Item, and b) making a statement about the rights-related aspect of the Item or part(s) of the Item to which rightsInfo applies.
2. There are two ways to express the scope:

2.1. In a general way: all elements of an Item are split into a) the set of metadata properties and b) the content. Thus it can be expressed that
- rightsInfo is about the Item as a whole by not having a @scope attribute
- rightsInfo is about the metadata properties only by adding a @scope attribute with a value of "riscope:metadata"
- rightsInfo is about the content only by adding a @scope attribute with a value of "riscope:content"
To see which parts of an Item fall under the content-scope, and which parts under the metadata-scope, check the definition in the Rights Info Scope NewsCodes.
When making a statement about the scope in this general way an @idrefs attribute MUST NOT be present on this rightsInfo element (else the scope will only apply to the element(s) with a corresponding @id).

2.2. In a specific way: by adding the ID(s) of XML element(s) to the @idrefs attribute this rightsInfo applies only to all element(s) which have a corresponding @id.
This specific addressing of elements overrides rightsInfo expressions which use the mechanism described in 2.1 above.
The application of rightsInfo is not inherited by the children of itemMeta and contentMeta if these wrapper elements are targeted using their IDs. Therefore their IDs should not be added to @idrefs.
If the referenced XML element is a partMeta element then:

- If a @scope attribute is not present then rightsInfo applies to both the content described by this partMeta element and to the metadata children of this partMeta element.
- If a @scope attribute is present its value(s) determines whether rightsInfo applies to the content described by this partMeta element or to the metadata children of this partMeta element.

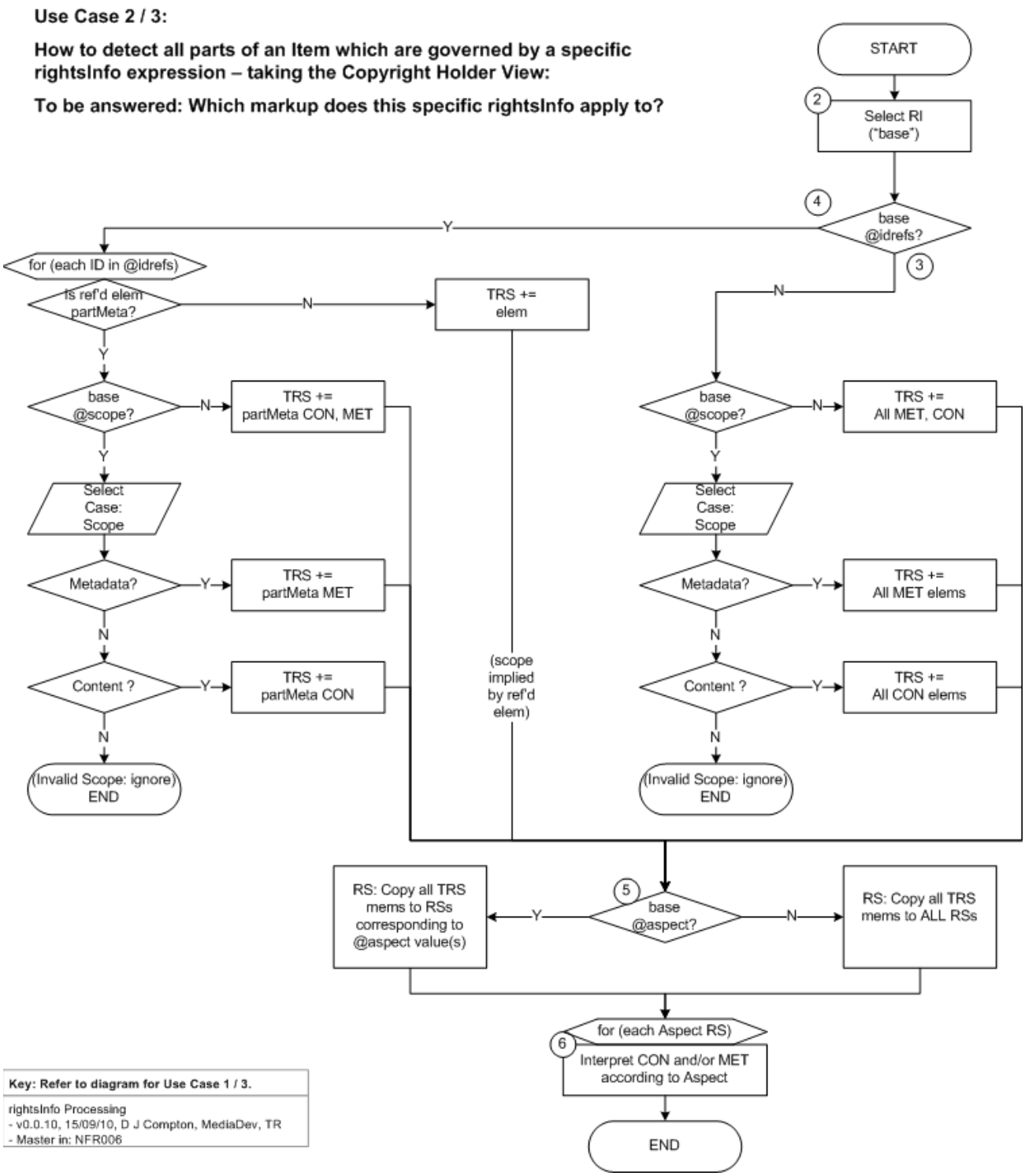
In compliance with the specification of the @idrefs attribute, IDs of only the following XML elements may be included into the list of values of @idrefs:
- all metadata properties as per the definition of the Rights Info NewsCode for "riscope:metadata".
- the child elements inlineXML, inlineData and remoteContent of contentSet of a News Item as they provide renditions of the full content, the child element concept of conceptSet of a Knowledge Item and the child element group of groupSet of a Package Item.

Explicitly excluded are all child elements of inlineXML of a News Item as they contain only parts of the content. In this case a partMeta element must be used to describe this part and the value of the @partid attribute of this partMeta element must be added to the list of values of the @idref attribute of the rightsInfo element.

3. The @scope and @idrefs attributes allow one to determine to which XML elements a rightsInfo element applies. In some cases it is necessary to associate a rightsInfo element with a particular aspect of an XML element. For example, a keyword element may contain a term associated with a photograph. One aspect of the keyword element to which a rightsInfo element may apply is the term itself. Another aspect to which a rightsInfo element may apply is the selection and application of this term to this photograph. Rights on these two aspects could be different. The @aspect attribute allows one to determine to which rights-related aspects the rightsInfo element applies.
If an @aspect attribute is not present then all aspects from the Rights Aspect NewsCodes apply.
If an @aspect attribute is present then only the aspects from the Rights Aspect NewsCodes listed in the attribute apply.
If a target does not support a specific aspect which is listed in the @aspect attribute then this aspect should be ignored for this target.

How to detect all parts of an Item which are governed by a specific rightsInfo expression - taking the Copyright Holder View:

To be answered: Which markup does this specific rightsInfo apply to?



1.The goal of the processing: the result will be multiple sets of elements and/or parts of content which all are governed by a rightsInfo expression. Each of the sets a) corresponds to one of the Rights Aspect NewsCodes, and b) may be empty after the processing if no corresponding parts of an item were found.

2. Select the rightsInfo element to be processed; this is the "base" for all subsequent processing steps.

3. If no @idrefs attribute exists in the base:

3.1. If a @scope attribute is not present: all the content and all metadata properties of this item are governed by the base's rights expression; they all should be included into a temporary result set. Continue with step 5.

3.2. If a @scope attribute is present:

3.2.1. If its value is "riscope:metadata": only metadata properties are in the scope of this rightsInfo element, add only all metadata elements of this item to a temporary result set. Continue with step 5.

3.2.2. If its value is "riscope:content": only content is in the scope of this rightsInfo element, add only all content of this item to a temporary result set. Continue with step 5.

4. If an @idrefs attribute is present in the base:

4.1. Iterate over each of the IDs listed by the @idrefs attribute and find the referenced element:

4.1.1. If the referenced element is a partMeta element then check if a @scope attribute is present in the base:

4.1.1.1. If a @scope attribute is not present: a) the partMeta content and b) all the partMeta metadata properties are governed by the base's rights expression; they all should be included into a temporary result set. Continue with step 5.

4.1.1.2. If a @scope attribute is present:

4.1.1.2.1. If its value is "riscope:metadata": only metadata properties are in the scope of this rightsInfo element, add only the metadata elements of this partMeta element to a temporary result set. Continue with step 5.

4.1.1.2.2. If its value is "riscope:content": only content is in the scope of this rightsInfo element, add only the content described by this partMeta element to a temporary result set. Continue with step 5.

4.1.2. If the referenced element is not a partMeta element: add the referenced element to a temporary result set. In this case the scope is implied by the element that is referenced and any @scope attribute should be ignored. Continue with step 5.

5. Check the base for an @aspect attribute:

5.1. If an @aspect attribute is not present then all members of the temporary result set should be copied to each of the result sets for the different Rights Aspects.

5.2. If an @aspect attribute is present then all members of the temporary result set should be copied only to the result sets corresponding to the Rights Aspects which are present in the @aspect list.

6. Final step: iterate over the result sets for the different Rights Aspects and interpret the included parts of the content or metadata elements according to the associated aspect. Some members of the result set may not be in a scope specified in the definition of the aspect; such members should be excluded from the result set.

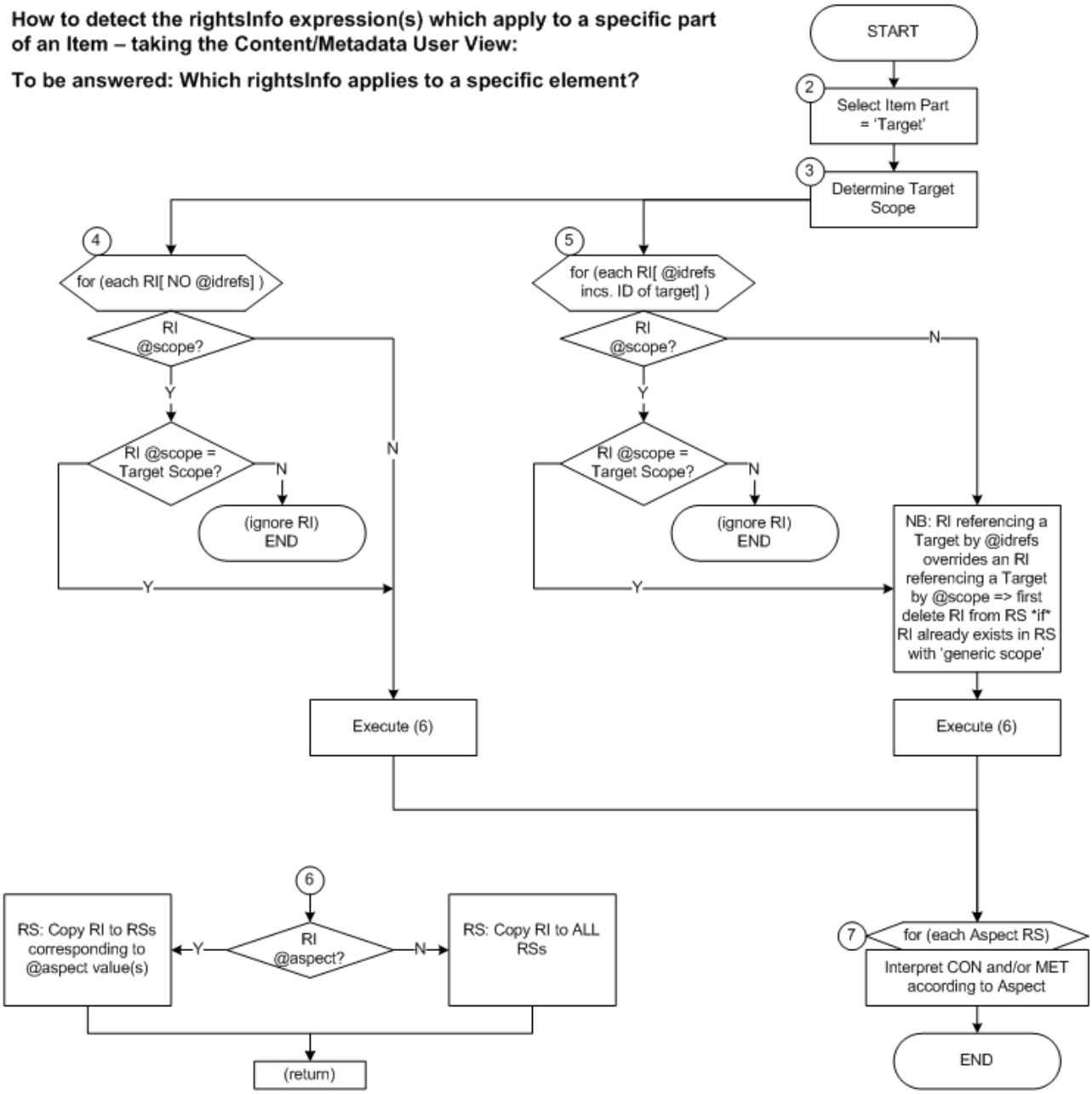
How to detect the rightsInfo expression(s) which apply to a specific part of an Item - taking the Content/Metadata User View:

To be answered: Which rightsInfo applies to a specific element?

Use Case 3 / 3:

How to detect the rightsInfo expression(s) which apply to a specific part of an Item – taking the Content/Metadata User View:

To be answered: Which rightsInfo applies to a specific element?



Key: Refer to diagram for Use Case 1 / 3.

ightsInfo Processing
v0.0.08, 15/09/10, D J Compton, MediaDev, TR.
Master in: NFR006

1. The goal of the processing: the result will be multiple sets of rightsInfo elements which all will apply to this part of the Item. Each of the sets a) correspond to one of the Rights Aspect NewsCodes, and b) may be empty after the processing if no corresponding rightsInfo elements were found.
2. Select the part of the Item for which the corresponding rightsInfo expression(s) should be determined, this part is the "target" for all subsequent processing steps.

This part must be a) the full content, b) one of the renditions of the content as a whole, c) a part of the content which is described by a partMeta element, or d) a single metadata property. The metadata wrappers itemMeta or contentMeta should not be selected as a target of this processing.

3. Define into which scope of rightsInfo elements the target falls:
   Match the target against the definitions of corresponding parts for "riscope:content" and "riscope:metadata" of the Rights Info Scope NewsCodes and determine to which scope the target belongs.
   Be aware that partMeta elements fall under both scopes.

4. Iterate over each rightsInfo element which has no @idrefs attribute:

4.1. If a @scope attribute is not present in the rightsInfo element then check the rightsInfo element against the rules of step 6 and add it to result sets as defined. Earmark the added rightsInfo element as "generic scope rightsInfo". Continue with step 7.

4.2. If a @scope attribute is present and the target falls in the scope of the attribute's value (see step 3) then check the rightsInfo element against the rules of step 6 and add it to result sets as defined. Earmark the added rightsInfo element as "generic scope rightsInfo". Continue with step 7.

5. Iterate over each rightsInfo element which has an @idrefs attribute that includes the ID of the target:

5.1. If a @scope attribute is not present then check this rightsInfo element against the rules of step 6. Be aware that a rightsInfo element which is referencing the target by @idrefs overrules rightsInfo elements which reference the target by @scope. For that reason if the target should be added to the result set then first delete any rightsInfo element which is earmarked as "generic scope rightsInfo" from the result set, and then add this rightsInfo element. Continue with step 7.

5.2. If a @scope attribute is present and the target falls in the scope of the attribute's value (see step 3) then check the rightsInfo element against the rules of step 6. Be aware that a rightsInfo element which is referencing the target by @idrefs overrules rightsInfo elements which reference the target by @scope. For that reason if the target should be added to the result set then first delete any rightsInfo element which is earmarked as "generic scope rightsInfo" from the result set, and then add this rightsInfo element. Continue with step 7.

6. Check any @aspect attribute of a rightsInfo element:

6.1. If an @aspect attribute is not present then the rightsInfo element should be added to the result sets corresponding to each of the Rights Aspect Newscodes.

6.2.If an @aspect attribute is present then the rightsInfo element should be added only to the result sets corresponding to the Rights Aspects which are present in the @aspect list.

7. Final step: iterate over the result sets for the different Rights Aspects and interpret the included parts of the content or metadata elements according to the associated aspect. Some members of the result set may not be in a scope specified in the definition of the aspect; such members should be excluded from the result set.

2.7 Item Metadata

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

2.7.1 Management Metadata

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

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

Table 1. Item Management Group Elements

Element Title	Element Name	Card	Described on Page
Item Class	**itemClass**	(1)	176
Content Provider	**provider**	(1)	102
Date Item Version Created	**versionCreated**	(1)	117
Date Item First Created	**firstCreated**	(0..1)	116
Date Item Embargo Ends	**embargoed**	(0..1)	115
Publish Status	**pubStatus**	(0..1)	225
Role in the Workflow	**role**	(0..1)	239
File Name	**filename**	(0..1)	149
Generator Tool	**generator**	(0..1)	152
Profile	**profile**	(0..1)	223
Editorial Service	**service**	(0..unbounded)	137
Item Title	**title**	(0..unbounded)	182
Editorial Note	**edNote**	(0..unbounded)	136
Member Of	**memberOf**	(0..unbounded)	192
Instance Of	**instanceOf**	(0..unbounded)	173
Signal	**signal**	(0..unbounded)	249
Alternative Representation	**altRep**	(0..unbounded)	72
Deliverable Of	**deliverableOf**	(0..1)	131
Hash Value	**hash**	(0..unbounded)	131

The IPTC provides a mandatory standardised scheme applicable to the **itemClass** (page 176) 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-G2 namespace. See also **XML Namespaces** (page 60) and **Extension Points in XML** (page 61).

2.7.2 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.

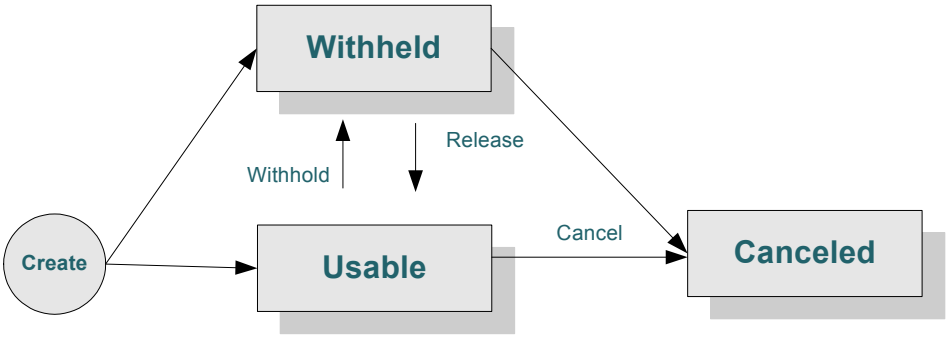
Embargoes are managed by the **embargoed** (page 115) property. At the level of the data model the embargoed element could be linked now to an **edNote** (page 136) element if the existing embargoed is empty (<embargoed />).

Details are described in the Processing model below

2.7.2.1 State Transition Diagram

This depicts the state transition diagram reflecting the ways in which the **pubStatus** (page 225) 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 1. State Transition Diagram



2.7.2.2 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).

2.7.2.3 Processing Model on the Recipient Side

Here is the processing model on the recipient side and relies on the **pubStatus** (page 225) 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 as described in **Table 2**.

Table 2. Test pubStatus = Usable

<embargoed>	<pubStatus>	How to Process
Element is absent.	Usable	Item is usable and not embargoed.
Element exists, provides a Date/Time value.	Usable	The embargo on the item ends at the given date and time.
Element exists, but is empty. Corresponding edNote exists.	Usable	The item is embargoed as long as a condition applies which is described in an editorial note. The edNote should be formulated like this: <edNote @role="noteRole:embargo">Until end of speech</edNote>
Element exists, but is empty. No corresponding edNote exists.	Usable	The item is embargoed indefinitely. This may be overridden by a contractual agreement between the provider and the client.

2.7.3 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[+-]HH:MM:SS") it MUST be considered as being not existent.

In the case of the mandatory **versionCreated** (page 117) property the full Item MUST be considered as being void.

2.7.4 Best Practice for expressing an update or correction of an item

An Update is expressed by using the concept URI http://cv.iptc.org/newscodes/signal/update (as QCode with the recommended scheme alias: sig:update) as value of the Signal property under the Item Meta of an Item. This signal indicates that some part of the item has been updated. This implies that this version of the item is not the inital version.

A Correction is expressed by using the concept URI http://cv.iptc.org/newscodes/signal/correction (as QCode with the recommended scheme alias: sig:correction) as value of the Signal property under the Item Meta of an Item. This signal indicates that some part of the item has been corrected. This implies that this version of the item is not the inital version. This Correction signal does not indicate in which version(s) of the item the corrected error existed.

In addition a concept from the Severity NewsCodes (http://cv.iptc.org/newscodes/severity/) may be used as a refinement

of how severe the impact of this update or change is. The IPTC acknowledges that the rules for applying the severity are set by the news provider of the item.

Further the Editorial Note (edNote) property under Item Meta may be used to provide details about the update or correction like pointing at a name in the text which has been corrected of if paragraph with updated information has been added to the text.

2.7.5 Best Practice for issuing a content warning

A Content Warning is expressed by using a QCode for the concept URI http://cv.iptc.org/newscodes/signal/cwarn with the **signal** (page 249) property. (With the recommended alias the QCode is "sig:cwarn".) This signal indicates that the content of the item should be reviewed as it may be perceived as being offensive.

In addition, refinement of the reason(s) for the content warning MAY be expressed by using concept(s) from the Content Warning NewsCodes http://cv.iptc.org/newscodes/contentwarning/ with the **exclAudience** (page 144) property.

Examples:

1. Content Warning signal without specific Content Warning NewsCodes:

```
<signal qcode="sig:cwarn"/>
```

2. Content Warning signal with specific Content Warning NewsCodes (relating to nudity and language):

```
<signal qcode="sig:cwarn"/>
<exclAudience qcode="cwarn:nudity"/>
<exclAudience qcode="cwarn:language"/>
```

2.8 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** (page 187) 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/itemrelation/.

To identify the target resource either the *residref* attribute or the href attribute MUST be set, optionally both MAY be used in parallel. The *residref* attribute identifies the target resource by its globally unique identifier (if the resource has such an identifier), while the *href* attribute identifies the location of the target resource in e.g. a (remote) file system. If the target resource is an Item and the *residref* attribute is used, a version attribute MAY indicate the target Item version; in the absence of version information, the target resource is the latest version available.

The content type, a.k.a. IANA MIME type of the target resource MAY be indicated by the contenttype attribute. It MAY be complemented by a format attribute to refine the MIME type information.

In order to ease the processing of a linked resource, the size in bytes of the target resource MAY be indicated. This feature is 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 **timeValidityAttributes** (page 307) (*validfrom* and *validto*) which express the date and time between which the link is valid.

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 data 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** (page 187) property (see the details for this inclusion in the **link** specification table for the "Hint and Extension Point" child elements).

2.8.1 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** (page 72)) 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.

2.9 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.

2.9.1 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)	259
Date Content Created	**contentCreated**	(0..1)	113
Date Content Modified	**contentModified**	(0..1)	114
Located	**located**	(0..unbounded)	189
Information Source	**infoSource**	(0..unbounded)	175
Creator	**creator**	(0..unbounded)	109
Contributor	**contributor**	(0..unbounded)	104
Audience	**audience**	(0..unbounded)	76
Excluded Audience	**exclAudience**	(0..unbounded)	144
Alternative Identifier	**altId**	(0..unbounded)	70

2.9.1.1 Dates Processing Model

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

contentCreated (page 113) and **contentModified** (page 114) 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[+-]HH:MM:SS”) it MUST be considered as being not existent.

DateValues.1: If contentCreated is present it MUST NOT be later than **versionCreated** (page 117).

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.

2.9.1.2 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** (page 76) and a set of "negative" values. 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.

2.9.2 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)	185
Genre	**genre**	(0..unbounded)	153
Keyword	**keyword**	(0..unbounded)	183
Subject	**subject**	(0..unbounded)	253
Slugline	**slugline**	(0..unbounded)	250
Headline	**headline**	(0..unbounded)	160
Dateline	**dateline**	(0..unbounded)	127
By	**by**	(0..unbounded)	80
CreditLine	**creditline**	(0..unbounded)	110
Description	**description**	(0..unbounded)	130

2.9.3 Other Content Metadata

Each provider may add a set of metadata properties which have to be defined in a non-IPTC-G2 namespace. See also **XML Namespaces** (page 60) and **Extension Points in XML** (page 61).

2.10 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** (page 211) 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** (page 164) - 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. If multiple icon elements are present they MUST represent the same visual content, only differentiated by rendition, contentType or format.

A section of a stream MAY be defined by a **timeDelim** (page 254) element. 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 **regionDelim** (page 261) element. Currently regions are limited to rectangles defined by {x, y, width, height} coordinates in pixels expressed as a set of attributes.

The role of this part in a stream of content MAY be defined by the **role** (page 238) property.

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 Administrative and Descriptive Metadata may be applied to each part, in complement to the administrative and 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 -G2 namespace. See also **XML Namespaces** (page 60) and **Extension Points in XML** (page 61).

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

2.10.1 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.

2.10.2 Time Unit Types and Start/End Timestamp Formats

The format of the Start Timestamp (@start) and/or End Timestamp (@end) is implied by the associated Time Unit type (@timeunit), see the Time Delimiter element - **timeDelim** (page 254).

Table 5 defines the processing of values of the three related attributes - but be aware: they are required by the XML Schema but may either show invalid values or be empty.

Table 5. Time Unit Type and Start/End Value Processing

Time Unit Type [@timeunit]	Start/End Timestamp [@start / @end]	How to Process
Invalid value	None	Ignore the Time Delimiter.
Invalid value	One or both	The default Time Unit Type value of editUnit MUST be used; the related format is used to parse the Timestamp value(s).
Valid value	None	Ignore the Time Delimiter.
Valid value	One or both	The defined Time Unit Type value MUST be used; the related format is used to parse the Timestamp value(s).

2.10.3 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** (page 74) property provides information about a concept identified by a qualified code. The information is given as a set of properties providing metadata about 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.

2.11 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** (page 169) 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.

If the concept is taken from a controlled vocabulary it MUST be identified by a qualified code, in any other case it SHOULD be identified by a literal value, and supplemental information MAY be given as a set of properties relative to the concept.

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.

2.12 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** (page 103) wrapper element. Their order of appearance in contentSet is of no relevance. Their presence is optional: this allows for a lightweight and extensible 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** (page 170) 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** (page 166) 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 content type must be set accordingly. Encoding is not constrained to base64 at this level of conformance.
- The **remoteContent** (page 233) 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 **timeValidityAttributes** (page 307) (*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.

All these three types of content component elements have an id attribute. For this attribute a special constraint applies: its value MUST be persistent for all versions of the item, i.e. for its entire lifecycle. The reason for this constraint is that G2 elements referencing a target G2 item may further point inside this item to reference a specific content component by its - persistent - id.

2.13 News Content Characteristics

newsContentCharacteristics (page 311) 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.

2.14 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** (page 83) element MAY therefore be defined as a child of a **remoteContent** (page 233) 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.

3 Introduction to EventsML-G2

EventsML-G2 is 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). The EventsML-G2 specifications build on the NewsML-G2 structural specifications and add a well defined functionality for conveying events.

To better understand the terminology the IPTC used for the G2-Standards specification we recommend the **Glossary** (page 315) as reference, as it provides an extensive set of terms and their definitions.

3.1 Overview

3.1.1 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 builds on common specification for the exchange of news items and knowledge about topics, concepts and events.
- EventsML-G2 may be used for:
  - Receiving all facts about a specific 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
  - Storing facts about knowledgeable events in archives to be referenced by other items

3.1.2 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 by concepts identified 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.

3.1.3 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 a “daybook”, 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.

3.2 Definitions

3.2.1 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.

3.2.2 Coverage Information (LEGACY)

Note: the G2-Standards have a new tool for expressing and managing the planned coverage of events: **Planning news coverage - planningItem** (page 48). To provide backward compatibility the structure for coverage information as part of an event structure is still valid, but it is strongly recommended to move to the new design.

The old-style 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.

3.2.3 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:

- Persisting Knowledge: is information which is remembered and referenced to for a long time.
- Topical News: is typically volatile information in the sense of “nothing is older than yesterday's news”.

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

- 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 Knowledge Items: a variant with event specific extensions is available, in particular event details are added to the concept structure inside the Knowledge Item. Knowledge Items may be used to exchange a set of event information if it should be distributed with a concept identifier.

  Find details about this data model in section **An Event in a Concept Item or Many Events in a Knowledge Item** (page 36).

- 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 38).

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.

3.3 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 60).

3.4 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 6. 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)

Table 6. iCalendar-to-EventsML-G2 Component Mapping (Continued)

iCalendar Event Component (Alphabetically)	Corresponding EventsML-G2 Component
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)
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.

4 Events

4.1 The Core Information about Events

Regardless whether the information about an event is topical or persistent (see **The Data Model** on page 32) 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 91) for the event. This name should be rather concise and could be expressed in different languages.
- A natural language **definition** (page 88) 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 201) 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.
- The properties **sameAs {Relationship}** (page 244), **broader** (page 79), **narrower** (page 194) and **related** (page 229) 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.

  A **related** property may be used to further qualify the nature of the event. Related can take either an arbitrary literal value or a value 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.

Then a set of rather event-specific properties - wrapped by the **eventDetails** (page 141) property:

- A **dates** (page 128) 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 RFC 2445 (see more below).
- An **occurStatus** (page 203) 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 228) 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 65) information.
- A set of **participationRequirement** (page 210) 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 253) properties to express what the event is about. Be aware of the difference between a related and a subject property: related 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 related concept, while "music" or "Wolfgang Amadeus Mozart" is a matching subject.
- A set of **location** (page 190) 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 209) 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 207) 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 95) 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 185) properties to reflect all languages which will be spoken at the event.
- A **newsCoverage {Concept}** (page 195) property is still present in the specifications, purely for backwards compatibility; be aware that its status has changed to DEPRECATED in EventsML-G2 1.6. Conveying information about the planned coverage of an event should now use the generic G2 **Planning news coverage - planningItem** (page 48).
- As for many wrapping elements in G2-Standards, the information about an event can also be extended by provider-specific properties.

4.2 Event Information in Items

4.2.1 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.

As for EventsML-G2 versions higher than 1.7 IPTC recommends to use only the string "NewsML-G2" for the standard attribute and the corresponding NewsML-G2 version as no formal distinction is made anymore between NewsML-G2 and EventsML-G2 specifications.

Samples:

```
<conceptItem standard="EventsML-G2" standardversion="1.7"
      guid="urn:newsml:iptc.org:20091231:eventsample" version="4"
      xmlns="http://iptc.org/std/nar/2006-10-01/" > .....
</conceptItem>
<newsItem standard="NewsML-G2" standardversion="2.9"
      guid="urn:newsml:iptc.org:20101231:newssample" version="2"
      xmlns="http://iptc.org/std/nar/2006-10-01/" > .....
</newsItem>
```

4.2.2 An Event in a Concept Item or Many Events in a Knowledge Item

The persisting knowledge facet of event information is represented by the Concept Item (see **Managing Individual Concepts - conceptItem** on page 43), and subsequently the Knowledge Item (see **Managing Sets of Concepts - knowledgeItem** on page 44).

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.

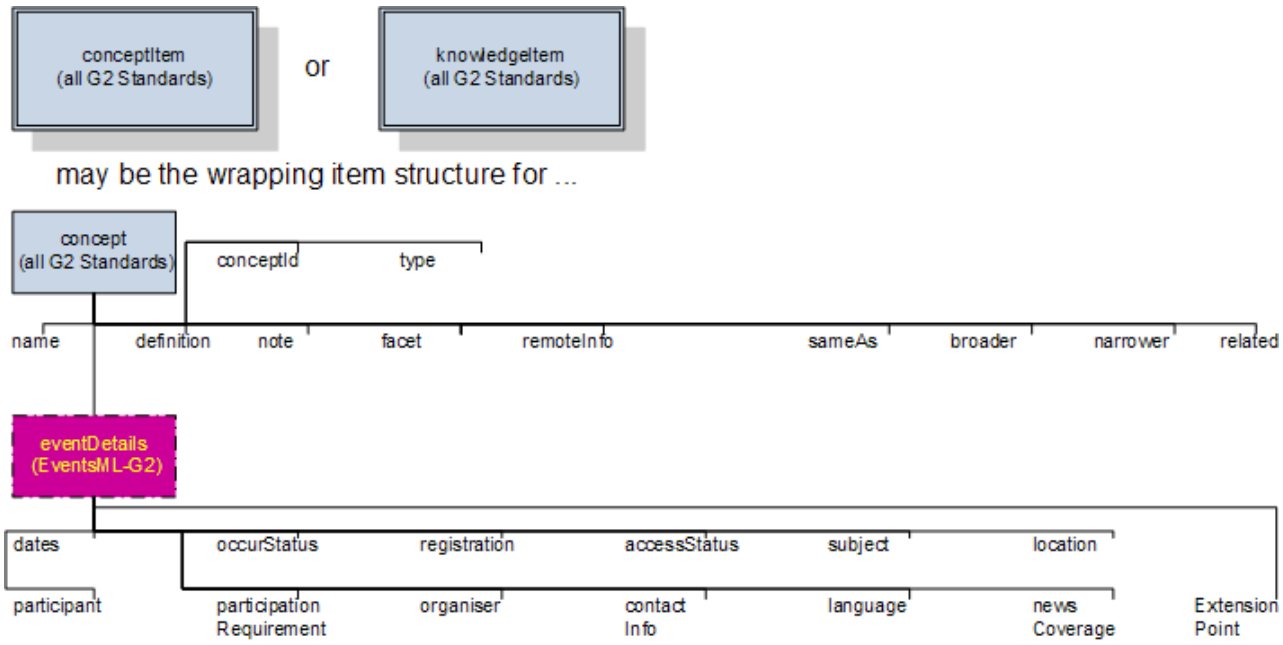
The generic part of a concept is used for event concepts too:

- The Concept Identifier for this event.
- Relationships to other events.
- A name, a definition, explanatory notes and refining related concepts.

Find more about the generic part in section **Representing Concept Information - concept Component** (page 40).

In Event Concept Items the value of the type of a concept (conceptItem/concept/type) must be set to the 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 **Same As** (page 244), **Broader** (page 79), **Narrower** (page 194)" and **Related Concept** (page 229) 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.

Knowledge Items with event concepts should be used to distribute event information if this information is planned to be updated - as this requires an identifier for each event.

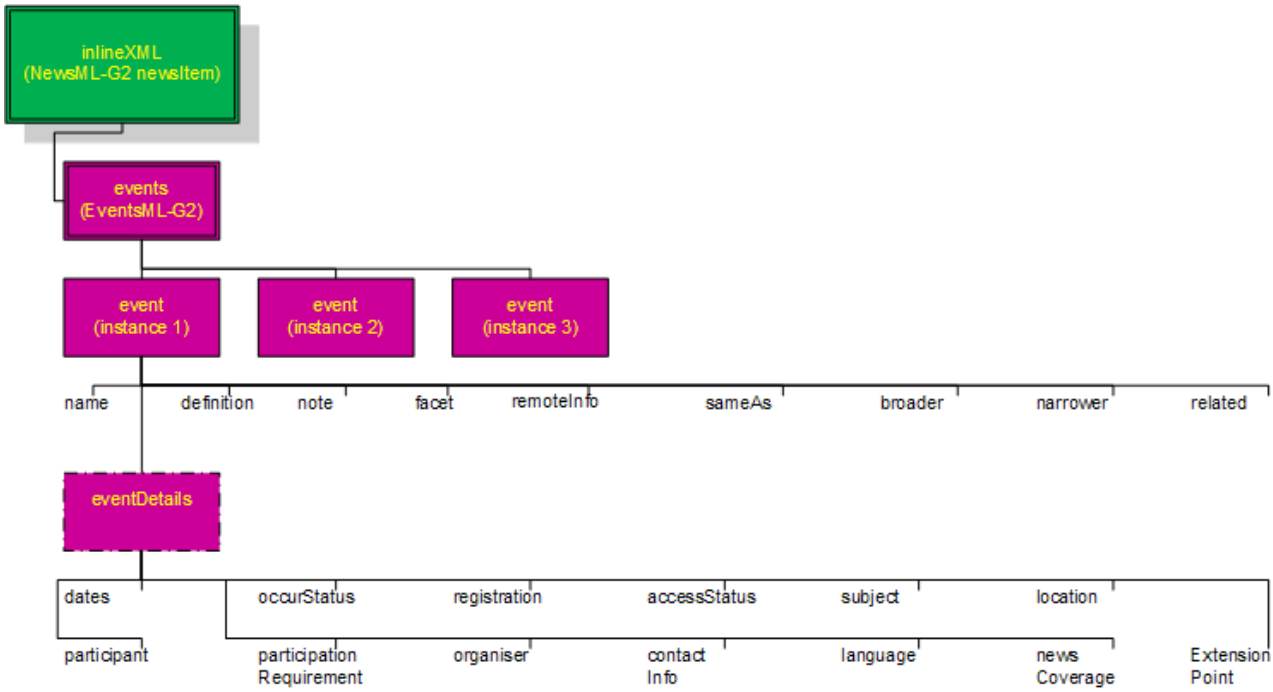
A provider could think of this use case scenario: he circulates a "top events of the next weekend" Knowledge Item with event concepts on Monday. On Wednesday he sends a new version of this Knowledge Item with updated events, and events that have been cancelled will be removed from the Knowledge Item.

4.2.3 Events in a NewsItem

Topical event information may be conveyed by using the NewsML-G2 NewsItem (see **Representing News - newsItem** on page 12) 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 143) element will be added as child to **Inline XML** (page 170) and it acts as a wrapper of one-to-many **Event** (page 140) elements, each representing the topical information of a single event.

Figure 3. 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.

4.2.3.1 News Metadata

In general the News Metadata section of a NewsItem - wrapped by the **Content Metadata {News Item}** (page 98) 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.

5 Representing Concept Information - concept Component

5.1 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, geopolitical areas, points of interest and objects for which a specific set of properties is defined for the purpose of a refined definition and improved search and processing capabilities.

The **concept** (page 86) 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 a **conceptId** (page 89) child element.

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** (page 258) child element. The type of a concept reflects its nature, e.g. generic, person, organisation, geopolitical area, point of interest etc…

A concept MUST have a **name** (page 224) and MAY be further defined by **remoteInfo** (page 236) and in natural-language by a **definition** (page 88) or **note** (page 201). These labels are repeatable and MAY be specified in multiple languages.

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** (page 224), **definition** (page 88), **note** (page 201), **remoteInfo** (page 236) MAY have *validfrom* and *validto* attributes which limit the use of the property in time.

The remoteInfo element MAY be used to express any external information about the concept as such. Be aware that the link element in the itemMeta wrapper should only be used for linking a Concept Item as a whole to another resource, e.g. a previous version, or another ConceptItem from which this one was derived and not to resources relevant to describing the Concept.

A **hierarchyInfo** (page 161) element MAY be used to express the location of this concept in the hierarchical tree of a taxonomy. For this purpose the hierarchyInfo holds a space separated sequence of the Concept Identifier QCodes of the ancestors of this concept, plus the Concept Identifier QCode of this concept. The sequence runs from left to right, with the top level QCode on the left, and the QCode of this concept on the right.

If the same concept is also defined in a different scheme this alternative identifier MAY be expressed by a **sameAs {Relationship}** (page 244) child element.

The sameAs element MUST have either a qcode or a literal attribute which identifies a concept, for the exact rules see the table below in the chapter **Relationships Between Concepts** (page 41). It MAY additionally have a type attribute which reflects the nature of the associated concept, and MAY have one or more names (see **Flexible 1 Property Type** (page 280). *validfrom* and *validto* attributes MAY limit the relationship in time.

More detailed properties of a concept (e.g. that the concept "is" an artist, listed company, city, restaurant) MAY be expressed by a specific **related** (page 229) property. The related property SHOULD have a rel attribute which specifies the exact relationship between this concept and the target concept (e.g. "is a",

"has a", "works for", "owns" ...). The IPTC provides a set of Concept Relationship NewsCodes for this purpose which is available at http://cv.iptc.org/newscodes/conceptrelation/.

5.2 Relationships Between Concepts

For any concept a relationship to another concept MAY be established, this may take 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 establish a set of the most standard relationships **broader** (page 79), and **narrower** (page 194) and further MAY add a more flexible **related** (page 229) relationship.

As the properties sameAs, broader, narrower and related establish a relationship to another property it is required to identify or describe this related concept. A specific selection out of three attributes MUST be used for this purpose. The basic rule as that all of them or none of them MUST NOT be used in any case. How the attributes MUST be used with the different properties for establishing a relationship is defined in the table below. (Be aware that establishing a relationship to an arbitrary value is special to the **related** (page 229) property only, the specification section of this element provides more details.)

Table 7. Why attributes to use with relationship properties

Property	Attribute qcode	Attribute literal	Set of attributes of an arbitrary value	Use case
sameAs	Yes	No	No	1
narrower	Yes	No	No	1
	No	Yes	No	2
broader	Yes	No	No	1
	No	Yes	No	2
related	Yes	No	No	1
	No	Yes	No	2
	No	No	Yes	3

Use cases for using the attributes to express the value to which the relationship should be established:

1) The value is a concept from a controlled vocabulary

2) The value is a concept which is not from a controlled vocabulary

3) The value is not a concept.

Further the sameAs, broader, narrower and related properties MAY have a type attribute which reflects the nature of the associated concept, and MAY have one or more names (see **Flex1PropType** (page 280)).

The *broader*, *narrower* and *related* properties MAY also have *validfrom* and *validto* attributes which limit the relationship in time, a *rel* attribute which details the name given to the relationship and a *rank* attribute which specifies the rank of the current concept among concepts having a relationship to the target concept. They also have a *facet* child property for expressing an intrinsic property of the related concept.

The *related* property has a *bag* property for allowing the expression of composite concepts (see **Composite Concepts** on page 55).

5.3 Details Associated with Specific Entities

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

personDetails (page 214) include a date of birth (born) and date of death (died) a repeatable indication of affiliation with an organisation and contact information (contactInfo).

organisationDetails (page 206) 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 (page 155) 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* attribute) expressed as a string. In the absence of this attribute, the WGS84 system is assumed.

POIDetails (page 218) include the geographic coordinates (position) and the postal address 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).

objectDetails (page 202) include a date of creation, a creator of the object and a copyright notice.

5.3.1 Contact Information

contactInfo (page 95) 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, postal addresses and notes.

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.

5.3.2 Postal Address

The definition of a Postal Address (**address** (page 219)) 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.

6 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).

6.1 Description

A **conceptItem** (page 90) 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.

6.2 Structure of a conceptItem

The model of a conceptItem is very similar to the model of a newsItem. Both share the same indicators 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** (page 89)) defined by an authority and conveyed as part of the content of the Item.

6.3 Item Metadata

The IPTC provides a mandatory standardised scheme applicable to the **itemClass** (page 176) property, identified by the URI: http://cv.iptc.org/newscodes/cinature/.

6.4 Concept related Metadata

The set of administrative metadata is common to all classes of Items. Please review **Representing News - newsItem** (page 12) for more information.

A conceptItem does not support descriptive metadata.

6.5 conceptItem Content

The content of a conceptItem is a concept component (see **Concept Component** on page 40).

7 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).

7.1 Description

A **knowledgeItem** (page 184) 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.

7.2 Structure of a knowledgeItem

The model of a knowledgeItem is very similar to the model of a newsItem. Both share the same indicators of compliance with a standard and conformance level, Identification and Versioning, Signature, Rights Information, Item Metadata, Item links. Please review **Representing News - newsItem** (page 12) for more information.

7.3 Item Metadata

The IPTC provides a mandatory standardised scheme applicable to the **itemClass** (page 176) property, identified by the URI http://cv.iptc.org/newscodes/cinature/.

7.4 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 8. Descriptive Metadata Core Group Elements

Element Title	Element Name	Card	Described on Page
Language	**language**	(0..unbounded)	185
Keyword	**keyword**	(0..unbounded)	183
Subject	**subject**	(0..unbounded)	253
Slugline	**slugline**	(0..unbounded)	250
Headline	**headline**	(0..unbounded)	160
Description	**description**	(0..unbounded)	130

Each provider may add a set of metadata properties which have to be defined in a non-IPTC-G2 namespace. See also **XML Namespaces** (page 60) and **Extension Points in XML** (page 61).

7.5 knowledgeItem Content

A **conceptSet** (page 93) wrapper element contains a set of concept components (see **Concept Component** on page 40). 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.

8 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** (page 208) 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.

8.1 Structure of a packageItem

The model of a packageItem is very similar to the model of a **newsItem** (page 199). Both share the same idicators 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.

8.2 Item Metadata

The IPTC provides mandatory 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/.

8.3 Package Related Metadata

The set of administrative and descriptive metadata is common between packageItems and newsItems. Please review **Representing News - newsItem** (page 12) for more information.

8.4 packageItem Content

A **groupSet** (page 158) 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** (page 156) 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 groupSet element MUST be the local reference to the group acting as a root.

A group component contains references to other group components (**groupRef** (page 157) with its *idref* attribute) of the same package item and references to Items or Web resource (**itemRef** (page 180) with its *guidref* and *href* attributes), in any order.

Each group MUST have an *id* attribute which identifies this group, and each group MUST 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. 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>
```

9 Planning news coverage - planningItem

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

9.1 Description

The **planningItem** (page 217) facilitates conveying the planning of news and topic coverage from the editorial of the news provider to the editorials of his clients. This item was introduced with the EventsML-G2 1.6 and NewsML-G2 2.7 (both based on the News Architecture version 1.8) and it is intended to replace the information about planned newscoverage provided as sub-structure of the Event Details of an Event Concept Item by EventsML-G2. As the Planning Items is part of the common News Architecture framework it can be used in the scope of EventsML-G2 and NewsML-G2.

Typical characteristics of a planningItem are:

- It focuses on planning and delivering the coverage of a single event or topic but may be linked to other planning items to facilitate the coverage of e.g. large or long-lasting events or a group of topics.
- It will usually be updated frequently until all planned coverage is delivered
- Its content is a structured representation of typical parameters of editorial planning and further may provide a list of G2 items which have been delivered to fulfil the intended coverage.
- It may refer to the event it covers: examples are media events like press conferences, political events like an election, cultural events like an open-air concert, or sport events.
- It may refer to the topic(s) it covers: examples are topics like "The current housing market", "The cultural festival summer season in Europe", "The best skiing resorts in the Rocky Mountains".

9.2 Structure of planningItem

The model of a planning item is very similar to the other G2 items: It shares the indicators of compliance with a standard and a conformance level, Identification and Versioning, Signature, Rights Information, Item Metadata and Item links. Please review **Representing News - newsItem** (page 12) for more information.

9.3 Item Metadata

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

9.4 Planning 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 9. Descriptive Metadata Core Group Elements

Element Title	Element Name	Card	Described on Page
Language	**language**	(0..unbounded)	185
Keyword	**keyword**	(0..unbounded)	183
Subject	**subject**	(0..unbounded)	253
Slugline	**slugline**	(0..unbounded)	250
Headline	**headline**	(0..unbounded)	160
Description	**description**	(0..unbounded)	130

Each provider may add a set of metadata properties which have to be defined in a non-IPTC-G2 namespace. See also XML Namespaces (page 48) and Extension Points in XML (page 49).

9.5 Metadata Helpers

The planningItem includes two properties which are available to help making metadata assertions:

- the assert property: helps to bundle and extend details of concepts, see **Assertions About Concepts** (page 28)

- the inlineRef property: helps to reference concepts which are inline of free text properties of type label, see **References to Inline Concepts** (page 28)

9.6 Planning Item Content

A **newsCoverageSet** (page 197) wrapper element contains a set of newsCoverage components (see below). Their order of appearance in conceptSet is not relevant. The major reason for having multiple newsCoverage components in this set is that each newsCoverage may be bound to e.g. a specific itemClass. Thus for including the coverage of an event by two text stories, 10 photos and one graphic one would have to used three newsCoverage components.

The **newsCoverage {Planning}** (page 196) component holds the mandatory planning property and the optional delivery property.

The **planning** (page 216) property - at least one has to be present in the wrapper - provides a rich set of properties which should tell the receiver what kind of coverage he can expect from the provider:

The g2contentType and the itemsClass properties tell what type of G2 deliverables to expect, and the itemCount adds how many of them to expect. The properties scheduled and service add when and by which service, or feed, the coverage will be delivered. A group of Descriptive Metadata gives a hint for the metadata which will be used with the delivered items allowing the receiver to build a filter or to forward this planning information to the proper desk of his editorial. The assignedTo property holds the person, organisation or company who has to deliver the content, this property can be of internal use of the news provider only or may be used to let the receivers know that e.g. a well-known journalist will write a review of a cultural event. If anything cannot be expressed by these machine readable properties the edNote may be used to add more information by natural language.

The **delivery** (page 133) property can add what of the planned coverage has been delivered. It is a wrapper for a set of deliveredItemRef properties, each of them pointing to a G2 property which has been delivered.

Be aware that the itemMeta wrapper of all G2 items includes a **deliverableOf** (page 131)property. This property can be considered to be a link back to this Planning Item and a specific News Coverage component of it. By these means the receiver can check by using the deliveredItemRef properties if an item indicated as "being delivered" has already arrived. On the other hand if a G2 item being a deliverable of a planned coverage arrives before an updated version of the planningItem has arrived already indicates that it pertains to a specific editorial planning. A news provider should take care to keep these two links in synch.

9.7 Processing Considerations

It can be expected that Planning Items will have a high frequency of updates. The first version may be sent when the first outline of covering an event or a topic has been completed by the editorial of the news provider. Updates could and should be sent when types of planned G2 items extend (e.g. first text only is planned, later extended to text plus photos), when the number of planned items change or when typical metadata values for the items have been assigned. In the course of creating and delivering the items the Planning Item may be updated each time such an item (or a group of items, like a group of graphics) has been released.

10 Dealing with Controlled Values

10.1 {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 51) 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 by a Catalog (see **Catalog of Controlled Vocabularies** on page 13) 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.

10.2 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** (page 176) or **pubStatus** (page 225).

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 **QCodeType** (page 297) 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.

10.2.1 Lexical Space Specification and Processing Model for Scheme URIs, Scheme Aliases, Codes, and QCodes

10.2.1.1 Lexical Space

- Lexical space for Scheme URIs: conforms with the Unreserved Characters of RFC 3986, section 2.3. Reserved Characters as per RFC 3986, section 2.2 must be considered depending on the selected URI scheme.
- Lexical space for Aliases: all characters except colon (#x3A) and white space (#x20 | #x9 | #xD | #xA).
- Lexical space for Codes:
  - All Unreserved Characters of RFC 3986, section 2.3., except white space (#x20 | #x9 | #xD | #xA). Reserved Characters as per RFC 3986, section 2.2 must be considered depending on the selected URI scheme.
  - Whitespace characters must be replaced by a sequence of one or more unreserved characters that is reused for this purpose according to the practices of the provider; it is recommended that such a sequence is not part of the any of the codes used by the provider

10.2.1.2 Processing Model

Given are:

- a Scheme URI
- a corresponding Scheme Alias
- a Code

10.2.1.2.1 QCodes

10.2.1.2.1.1 Creating QCodes (any G2 Item)

1. Concatenate the Scheme Alias, a colon and the Code to a string.
2. Apply any required XML encoding to this string (Note: this is typically done by the XML processor software).
3. The resulting string is applied as a QCode value.

10.2.1.2.1.2 Receiving/Parsing QCodes from an XML Document (any G2 Item)

1. Retrieve the QCode value from the XML document
2. Apply any required XML decoding (Note: this is typically done by the XML processor software) .
3. To split a QCode term into the alias and the code part, identify the first colon searching from left to right. The string to the left of the colon is the alias; the string to the right is the code. If it encounters none the QCode is invalid.
4. Check whether the alias is defined in the catalog. If it is not, the QCode is invalid

10.2.1.2.2 Concept URIs

G2 processors should be able to process Internationalized Resource Identifiers (IRIs) as per RFC 3987.

10.2.1.2.2.1 Creating a Concept URI/IRI:

1. Concatenate the Scheme URI and the Code resulting in the Concept URI.
2. Check if any Reserved Characters (see RFC 3986 section 2.2) are included based on the used URI scheme. In particular check for the reserved characters of the http scheme. If such characters are used apply percent-encoding as per RFC 3986.

Note: This check and any corresponding changes should be done by software at the application level because only at this level it is known if e.g. a & character is an http URL delimiter or a character in one of the parts of the URI.

3. With this (potentially encoded) string:

First, consider it from the IRI RFC 3987, section 3.1 perspective:

When the IRI is used solely for identification purposes, it is not necessary to map the IRI to a URI (see section 5). However, when an IRI is used for resource retrieval, the resource that the IRI locates is the same as the one located by the URI obtained after converting the IRI according to the procedure defined here.

Second, if the IRI is an IRI as per RFC 3986 and if it should be used for retrieval, apply any required percent-encoding as defined by RFC 3987 and RFC 3986. This encoding could be done by standard software (component).

10.2.1.2.2.2 Comparing Concept URIs/IRIs:

If provided Concept URIs are IRIs per RFC 3987 then they must be compared for equivalence as defined per RFC 3987, section 5.

If provided Concept URIs are URIs per RFC 3986 then they must be compared for equivalence as defined per RFC 3986, section 6.

10.3 Processing Catalogs

10.3.1 Structure of a Catalog

A **Catalog** (page 82) MUST have one or more scheme elements. A catalog MAY have one or more titles in different languages. It MAY also have a pointer to some additional information available on the Web, and especially its evolution and latest version. Such information will help people follow the evolution of a shared catalog like the IPTC G2 catalog, and include in their Items a reference to the latest version if they wish.

It 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.

10.3.2 Processing Remote Catalogs

By activating the hyperlink of a remoteCatalog (see **catalogRef** on page 232), a plain catalog structure is returned, and MUST be processed as if were locally defined.

10.3.3 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}.

10.3.4 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.

10.4 Processing Schemes

10.4.1 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.

10.4.2 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.

10.5 Qualified and Typed Properties

Qualified properties – of datatype **QualPropType** (page 298) – 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 **TypedQualPropType** (page 304) – 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** (page 109), **contributor** (page 104) and **infoSource** (page 175) 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.

10.6 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 that literal identifiers or no identifiers at all to be applied instead of controlled identifiers. Additionally, a free-text value in the literal attribute is an identifier of a concept and NOT a human readable description. Therefore flexible properties - of datatype Flexible Property Type or a derived datatype - support both controlled (qcode) and uncontrolled (literal) identifiers - or no identifier at all.

QCodes and literals are mutually exclusive for any given property; if one of them exists the other one MUST NOT exist.

The rules for using the qcode or the literal attribute or no concept-identifying attribute at all with a property are:

- If a bag is used with a property then qcode and literal attributes MUST NOT be used with the property.

- If a bag is not used with a property then the property MAY have a qcode attribute OR a literal attribute or neither.

- If a literal value is used with an assert property then all instances of that literal value in that item MUST identify the same concept.

- If a literal value is not used with an assert property then it is NOT required that all instances of that literal value in that item identify the same concept.

Literals MAY be used in the following cases:

1) When a code from a vocabulary which is known to the provider and the recipient is used without a reference to the vocabulary: The details of the vocabulary are communicated outside of the G2-Standards specifications. Such a contract could express that a specific vocabulary of literals is used with a specific property.

2) When importing metadata: The values of literals may contain codes which have not yet been checked to be from an identified vocabulary.

3) As an identifier for linking with an assert element: The value could be a random one. If a literal value is used with an assert property then all instances of that literal value in that item must identify the same concept.

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. If the value of the property is a literal and no additional name is given, the recipient MAY use the literal value for direct display. But as the primary use of a literal is being an identifier it may not tell much about the meaning of the metadata.

Flexible properties MAY also be complemented by other information about the concept, like properties from Concept Relationships Group (see **Table 200** on page 263) and Concept Definition Group (see **Table 199** on page 263).

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.

10.7 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** (page 278) – 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** (page 77) 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}

10.8 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 (see **Table 245** on page 306), 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.

11 Dealing with Labels and Blocks

11.1 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** (page 73)) for the inclusion of hyperlinks, the **span** (page 251) as a generic mechanism for adding information to text, simple **ruby** (page 240) markup used in Japanese publications and **inline** (page 165) 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** (page 74) property value, plus a basic set of properties defining the concept.

11.2 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 (see **Table 244** on page 306) , i.e. an optional language tag (xml:lang) and indication of the directionality of textual content (dir).

12 Exchanging Items - newsMessage

A **newsMessage** (page 200) 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** (page 181) 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.

12.1 Message Information

All the information about the newsMessage as a wrapper of conveyed G2 items is collected under the **header** (page 193) element which MUST be present.

A newsMessage MUST have a date of transmission – **sent** (page 126). The date of transmission MAY not be updated in case of retransmission of the message.

f any QCode is used within the header then a catalog and/or a catalogRef property MUST be included in the header. The scope of the scheme elements of the local and/or remote catalog(s) is limited to the header element and its descendants.

A newsMessage MAY have a **sender** (page 248) 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** (page 257) – 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 have a **priority** (page 222) property to control the overall message transmission process.

It MAY indicate the point of **origin** (page 205) of the message, using a provider defined syntax.

It MAY have one or more **timestamp** (page 256)(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** (page 134) properties using a provider defined syntax, and the indication of one or more channels – **channel** (page 85) – of transmission.

It MAY have one or more **signal** (page 249) properties to instruct the news message processor that the content requires a specific handling.

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

12.2 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.

13 Specification Reference

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

13.1 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 LabelType 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.

13.2 General Specifications

13.2.1 XML Namespaces

Table 10. 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.

13.2.2 MIME Types

Table 11. MIME Types

MIME Type Identifier	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.
application/vnd.iptc.g2.planningitem+xml	For all kinds of G2 Planning Items.
application/vnd.iptc.g2.newsmessage+xml	For the G2 News Message

13.2.3 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 60).

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 60). 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, see below.

13.2.4 Hint and Extension Points in XML

To act as an Extension Point properties from any other XML namespace than the News Architecture for G2 namespace may be added.

To act as an Hint Point properties from the News Architecture for G2 namespace may be added.

The purpose of properties from the NAR namespace is to add a set of hints, i.e. properties which have to comply with the structure of the G2 item target resource but do not have to be extracted from it. These properties must be added this way:

- Immediate child properties of <itemMeta>, <contentMeta>, or <concept> - optionally with their descendants - may be used directly under the extension point

- All other properties require the full path excluding only the item's root element.

13.3 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).

13.4 Processing Model Terminology

For many components of the G2-Standards 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".

13.5 Component Structure Format

Table 12 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 12**.

Table 12. 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). For equally named elements an annotiation in curled brackets like {POI} provides a hint for the context of the element.
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.

Table 12. Component Structure Format

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.

13.6 Element Definitions

13.6.1 Access

Table 13. 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 269)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.2 Access Status

Table 14. 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 296)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.3 Accountable Person

Table 15. 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 286)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.4 Action in Hop History

Table 16. Action in Hop History

(XML) Data Model	Element
Namespace (prefix)	nar
Name	action
Title	Action in Hop History
Definition	An action which is executed at this hop in the hop history.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**QCode Property Type** (page 296)
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes CV for the target attribute: http://cv.iptc.org/newscodes/hopactiontarget/
Attribute(s)	▪ target (0..1); **QCode Type** (page 297); The target of the action in a content object. If the target attribute is omitted the target of the action is the whole object.
	▪ timestamp (0..1); XML Schema DateTime; The date and optionally the time (with a time zone) when this action was performed on the target.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.5 Address Line

Table 17. Adress 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	**International String Type** (page 290)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.6 Affiliation

Table 18. 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 283)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 307)	**Name**	**Datatype**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.7 Alternative Identifier

Table 19. 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 290).
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ type (0..1); **QCodeType** (page 297); A qualifier which indicates the context within which the alternative identifier has been allocated.
	▪ environment (0..1); **QCodeListType** (page 295); A qualifier which indicates the business environment in which the identifier can be used to access the content
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.8 Alternative Locator

Table 20. Alternative Locator

(XML) Data Model	Element
Namespace (prefix)	nar
Name	altLoc
Title	Alternative Locator
Definition	An alternative location of the asset representing the content.
User Note(s)	If there is more than one alternative locator, they SHOULD be qualified using the type attribute to distinguish between different identification schemes.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**IRIType** (page 291)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ type (0..1); **QCodeType** (page 297); A qualifier which indicates the context within which the alternative locator has been allocated.
	▪ role (0..1); **QCodeType** (page 297); A refinement of the semantics or business purpose of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.9 Alternative Representation

Table 21. 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 291)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ representation (0..1); **QCodeType** (page 297); A qualifier which specifies the way the targetItem is represented at this location.		
	▪ contenttype (0..1); XML Schema string; The IANA (Internet Assigned Numbers Authority) MIME type of the target resource.		
	▪ format (0..1); QCodeType; A refinement of a generic content type (i.e. IANA MIME type).		
	▪ size (0..1); XML Schema nonNegativeInteger; The size in bytes of the target resource.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **timeValidityAttributes** (page 307)	**Name**	**Datatype**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.10 Anchor

Table 22. 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.		
	▪ href (0..1); **IRIType** (page 291); 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 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **span** (page 251) (0..unbounded)		
	▪ **ruby** (page 240) (0..unbounded)		
	▪ **inline** (page 165) (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)			

13.6.11 Assertion

Table 23. Assertion

(XML) Data Model	Element
Namespace (prefix)	nar
Name	assert
Title	Assertion
Definition	An assertion about a concept; may include many details.
User Note(s)	The assertion about the concept may be used to merge multiple occurrences of concept details in properties into a single place or to extend the details of an assertion beyond the limited details other properties can provide.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ qcode (0..1); **QCodeType** (page 297); A qualified code identifying the concept the assertion is made about. Or ▪ literal (0..1); XML Schema normalizedString; A free-text value identifying the concept the assertion is made about. The use of qcode and literal is mutually exclusive, one of them MUST be used.

	Name	Datatype
▪ **editAttributes** (page 306)	id (0..1)	XML Schema ID
	creator (0..1)	QCodeType
	modified (0..1)	DateOptTimeType
▪ **i18nAttributes** (page 306)	Name	Datatype
	xml:lang (0..1)	XML Schema language
	dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Child Element(s)	▪ Hint and Extension Point (0..unbounded). Properties from the NAR namespace or from another XML namespace may be added. The purpose of properties from the NAR namespace is to add a set of hints, i.e. properties which have to comply with the structure of the G2 item target resource but do not have to be extracted from it. These properties must be added this way: - Immediate child properties of <itemMeta>, <contentMeta>, or <concept> - optionally with their descendants - may be used directly under the extension point - All other properties require the full path excluding only the item's root element.
XML Schema Note(s)	
Example(s)	

13.6.12 Assigned To

Table 24. Assigned To

(XML) Data Model	Element
Namespace (prefix)	nar
Name	assignedTo
Title	Assigned To
Definition	The party which is assigned to cover the event and to produce the planned G2 item
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PartyPropType** (page 279)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.13 Audience

Table 25. 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 268)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.14 Bag

Table 26. 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)	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **bit** (page 78) (1..unbounded)		
XML Schema Note(s)			
Example(s)			

13.6.15 Bag Item

Table 27. 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 296)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ significance; (0..1); Int100Type; Indicates how significant the event expressed by a bit of event concept type is to the concept expressed by this bit The scope of this relationship is limited to the bits of a single bag. See also the note below the table.
Child Element(s)	
XML Schema Note(s)	
Note & Example(s)	**Note** on the significance attribute: This attribute is assigned to a special use case of a bag with subject properties: the bag includes one bit representing an event and one or more other bits representing entities which are related to this event. Only in this case the significance attribute may be used to express the significance of this event to the concept of the bit carrying this attribute. If the bag includes more than one event, any significance attribute of bits in the bag SHALL be ignored. **Example** 1: A merger of two companies which is differently significant to the two parties of the merger: the significance of the merger for the small company is high while it is low to the global player company. <bag> <bit type="cpnat:event" qcode=" abevents:Merger123AB"/> <bit type="cpnat:organisation" qcode="isin:TinyCompany" significance="100"/> <bit type="cpnat:organisation" qcode="isin:GlobalPlayerCompany" significance="10"/> </bag>

13.6.16 Broader

Table 28. 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 300)
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)	

13.6.17 By

Table 29. 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 292)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.18 Capacity

Table 30. 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 292)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.19 Catalog

Table 31. 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)	▪ additionalInfo (0..1); **IRIType** (page 291); A pointer to some additional information about the Catalog, especially its evolution and latest version.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)	▪ **title** (page 182) (0..unbounded)		
	▪ **scheme** (page 247) (1..unbounded)		
XML Schema Note(s)			
Example(s)			

13.6.20 Channel of Remote Content

Table 32. 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 32. Channel of Remote Content (Continued)

Attribute(s)	▪ chnlid (0..1); XML Schema positiveInteger; A logical identifier of the channel		
	▪ type (0..1); **QCodeType** (page 297); The media type of the data conveyed by the channel.		
	▪ role (0..1); **QCodeType** (page 297); The role the data of this channel plays in the scope of the full content.		
	▪ **newsContentCharacteristics** (page 311)	**Name**	**Datatype**
		Word Count	XML Schema nonNegativeInteger
		Image Width	XML Schema nonNegativeInteger
		Image Width Unit	QCodeType
		Image Height	XML Schema nonNegativeInteger
		Image Height Unit	QCodeType
		Image Orientation	XML Schema nonNegativeInteger
		Image Colour Space	QCodeType
		Resolution	XML Schema positiveInteger
		Duration	XML Schema nonNegativeInteger
		Unit of Duration	QCodeType
		Audio Codec	QCodeType
		Audio Bit Rate	XML Schema positiveInteger
		Audio Variable Bit Rate flag	XML Schema boolean
		Audio Sample Size	XML Schema positiveInteger
		Audio Sample Rate	XML Schema positiveInteger
		Audio Channels	QCodeType
		Video Codec	QCodeType
		Video Average Bit Rate	XML Schema positiveInteger
		Video Variable Bit Rate flag	XML Schema boolean
		Video Frame Rate	XML Schema positiveInteger
		Video Scan Technique	enumeration progressive/interlaced
		Video Aspect Ratio	XML Schema nomalizedString
		Video Sampling Method	XML Schema nomalizedString
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.21 Channel for News Message

Table 33. 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)	If both are present the @literal and the property string value SHOULD be identical. If both are present but not identical @literal takes precedence		
XML Schema Spec	At: Both CCL and PCL		
Datatype	XML Schema string		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **qualifyAttributes** (page 309)	**Name**	**Datatype**
		qcode (0..1)	QCodeType
		literal (0..1)	XML Schema normalizedString
		type	QCodeType
		role	QCodeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.22 Concept

Table 34. Concept

(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: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ id (0..1); XML Schema ID; The local identifier of the property		
	▪ **i18nAttributes** (page 306)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **conceptId** (page 89) (1)		
	▪ **type** (page 258) (0..1)		
	▪ **name** (page 91) (1..unbounded)		
	▪ **definition** (page 88) (0..unbounded)		
	▪ **note** (page 201) (0..unbounded)		
	▪ **facet** (page 147) (0..unbounded)		
	▪ **remoteInfo** (page 236) (0..unbounded)		
	▪ **hierarchyInfo** (page 161) (0..unbounded)		
	▪ **Concept Relationships Group** (page 263) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	79
		narrower (0..unbounded)	194
		related (0..unbounded)	229
		sameAs {Relationship} (0..unbounded)	244
	▪ **Entity Details Group** (page 263) (0..1)	**Element Name**	**Page**
		geoAreaDetails (1)	155
		organisationDetails (1)	206
		personDetails (1)	214
		POIDetails (1)	218
		objectDetails (1)	202
	▪ **eventDetails** (page 141) (0..1)		
	▪ Extension Point (0..1). Any set of provider-defined properties.		

Table 34. Concept (Continued)

XML Schema Note(s)	
Example(s)	

13.6.23 Concept Definition

Table 35. 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 269)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 307)	**Name**	**Datatype**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.24 Concept Identifier

Table 36. 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 270)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.25 Concept Item

Table 37. 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 266)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta {Concept}** (page 96) (0..1)
	▪ **assert** (page 74) (0..unbounded)
	▪ **inlineRef** (page 169) (0..unbounded)
	▪ **concept** (page 86) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.26 Concept Name

Table 38. 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 290).
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes CV for the *part* attribute: http://cv.iptc.org/newscodes/namepart/
Attribute(s)	▪ role (0..1); **QCodeListType** (page 295); A refinement of the semantics of the name.
	▪ part (0..1); **QCodeType** (page 297); Specifies which part of a full name this property provides.
	▪ **timeValidityAttributes** (page 307) — **Name**: validfrom (0..1), validto (0..1); **Datatype**: DateOptTimeType, DateOptTimeType
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.27 Concept Reference

Table 39. Concept Reference

(XML) Data Model	Element
Namespace (prefix)	nar
Name	conceptRef
Title	Concept Reference
Definition	A reference to a target concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1ConceptPropType** (page 278)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.28 Concept Set

Table 40. 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 86) (0..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.29 Confirmation

Table 41. 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 296)
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)	

13.6.30 Contact Information

Table 42. 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 295); A refinement of the semantics of the set of contact information.
Child Element(s)	▪ **email** (page 138) (0..unbounded)
	▪ **im** (page 174) (0..unbounded)
	▪ **phone** (page 215) (0..unbounded)
	▪ **fax** (page 148) (0..unbounded)
	▪ **web** (page 262) (0..unbounded)
	▪ **address** (page 219) (0..unbounded)
	▪ **note** (page 201) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.31 Content Metadata {Concept Item}

Table 43. Content Metadata for a Concept Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta {Concept}		
Title	Content Metadata for a Concept Item		
Definition	A set of metadata properties about the content of 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 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **icon** (page 164) (0..unbounded); If multiple icon elements are present within a single contentMeta or partMeta property they MUST represent the same visual content, only differentiated by rendition, contentType or format.		
	▪ **Administrative Metadata Group** (page 264) (0..1)	**Element Name**	**Page**
		audience (0..unbounded)	76
		contributor (0..unbounded)	104
		creator (0..unbounded)	109
		contentCreated (0..1)	113
		contentModified (0..1)	114
		located (0..unbounded)	189
		infoSource (0..unbounded)	175
		urgency (0..1)	259
		exclAudience (0..unbounded)	144
		altId (0..unbounded)	70
	▪ **Descriptive Metadata Core Group** (page 264) (0..1)	**Element Name**	**Page**
		description (0..unbounded)	130
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.6.32 Content Metadata {Knowledge Item}

Table 44. Content Metadata for a Knowledge Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta {Knowledge}		
Title	Content Metadata for a Knowledge Item		
Definition	A set of metadata properties about the content of 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 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **icon** (page 164) (0..unbounded); If multiple icon elements are present within a single contentMeta or partMeta property they MUST represent the same visual content, only differentiated by rendition, contentType or format.		
	▪ **Administrative Metadata Group** (page 264) (0..1)	**Element Name**	**Page**
		audience (0..1)	76
		contributor (0..unbounded)	104
		creator (0..unbounded)	109
		contentCreated (0..1)	113
		contentModified (0..1)	114
		located (0..unbounded)	189
		infoSource (0..unbounded)	175
		urgency (0..1)	259
		exclAudience (0..unbounded)	144
		altId (0..unbounded)	70
	▪ **Descriptive Metadata Core Group** (page 264) (0..1)	**Element Name**	**Page**
		description (0..unbounded)	130
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.6.33 Content Metadata {News Item}

Table 45. Content Metadata for a News Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta {News}		
Title	Content Metadata for a News Item		
Definition	A set of metadata properties about the content of a News Item.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **icon** (page 164) (0..unbounded); If multiple icon elements are present within a single contentMeta or partMeta property they MUST represent the same visual content, only differentiated by rendition, contentType or format.		
	▪ **Administrative Metadata Group** (page 264) (0..1)	**Element Name**	**Page**
		audience (0..unbounded)	76
		contributor (0..unbounded)	104
		creator (0..unbounded)	109
		contentCreated (0..1)	113
		contentModified (0..1)	114
		located (0..unbounded)	189
		infoSource (0..unbounded)	175
		urgency (0..1)	259
		exclAudience (0..unbounded)	144
		altId (0..unbounded)	70
	▪ **Descriptive Metadata Group** (page 265) (0..1)	**Element Name**	**Page**
		by (0..unbounded)	80
		creditline (0..unbounded)	110
		dateline (0..unbounded)	127
		description (0..unbounded)	130
		genre (0..unbounded)	153
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.6.34 Content Metadata {Package Item}

Table 46. Content Metadata for a Package Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta {Package}		
Title	Content Metadata for a Package Item		
Definition	A set of metadata properties about the content of a Package Item.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Table 46. Content Metadata for a Package Item (Continued)

Child Element(s)	▪ **icon** (page 164) (0..unbounded); If multiple icon elements are present within a single contentMeta or partMeta property they MUST represent the same visual content, only differentiated by rendition, contentType or format.		
	▪ **Administrative Metadata Group** (page 264) (0..1)	**Element Name**	**Page**
		audience (0..unbounded)	76
		contributor (0..unbounded)	104
		creator (0..unbounded)	109
		contentCreated (0..1)	113
		contentModified (0..1)	114
		located (0..unbounded)	189
		infoSource (0..unbounded)	175
		urgency (0..1)	259
		exclAudience (0..unbounded)	144
		altId (0..unbounded)	70
	▪ **Descriptive Metadata Group** (page 265) (0..1)	**Element Name**	**Page**
		by (0..unbounded)	80
		creditline (0..unbounded)	110
		dateline (0..unbounded)	127
		description (0..unbounded)	130
		genre (0..unbounded)	153
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.6.35 Content Metadata {Planning Item}

Table 47. Content Metadata for a Planning Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	contentMeta {Planning}		
Title	Content Metadata for a Planning Item		
Definition	A set of metadata properties about the content of a Planning Item		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **icon** (page 164) (0..unbounded); If multiple icon elements are present within a single contentMeta or partMeta property they MUST represent the same visual content, only differentiated by rendition, contentType or format.		
	▪ **Administrative Metadata Group** (page 264) (0..1)	**Element Name**	**Page**
		audience (0..unbounded)	76
		contributor (0..unbounded)	104
		creator (0..unbounded)	109
		contentCreated (0..1)	113
		contentModified (0..1)	114
		located (0..unbounded)	189
		infoSource (0..unbounded)	175
		urgency (0..1)	259
		exclAudience (0..unbounded)	144
		altId (0..unbounded)	70
	▪ **Descriptive Metadata Core Group** (page 264) (0..1)	**Element Name**	**Page**
		description (0..unbounded)	130
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.6.36 Content Provider

Table 48. 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 284)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.37 Content Set

Table 49. 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 170) (0..unbounded) Or ▪ **inlineData** (page 166) (0..unbounded) Or ▪ **remoteContent** (page 233) (0..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.38 Contributor

Table 50. 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	Extends **FlexPartyPropType** (page 284)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 295); A refinement of the semantics of the property.
	▪ jobtitle (0..1); **QCodeType** (page 297); The job title of the creator in the news provider organisation.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.39 Copyright Holder

Table 51. 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 284)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.40 Copyright Notice

Table 52. 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 resource.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**RightsLabelType** (page 301)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.41 Country

Table 53. 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 280)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.42 Country Area

Table 54. 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 280)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.43 Creator

Table 55. Creator

(XML) Data Model	Element
Namespace (prefix)	nar
Name	creator
Title	Creator
Definition	A party (person or organisation) which created the resource.
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	Extends **FlexPartyPropType** (page 284)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 295); A refinement of the semantics of the property.
	▪ jobtitle (0..1); **QCodeType** (page 297); The job title of the creator in the news provider organisation.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.44 Credit Line

Table 56. Credit Line

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	creditline		
Title	Credit Line		
Definition	A free-text expression of the credit(s) for the content.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	IntlStringType		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.45 Date an Object Ceased to Exist

Table 57. Date an Object ceased to Exist

(XML) Data Model	Element
Namespace (prefix)	nar
Name	ceasedToExist {Object}
Title	*Date an Object ceased to Exist*
Definition	The date (and optionally a time) on which the object ceased to exist.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.46 Date a Point Of Interest Ceased to Exist

Table 58. Date a Point Of Interest ceased to Exist

(XML) Data Model	Element
Namespace (prefix)	nar
Name	ceasedToExist {POI}
Title	*Date a Point of Interest ceased to Exist*t
Definition	The date (and optionally a time) on which the Point Of Interest ceased to exist.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.47 Date Content Created

Table 59. 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) on 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 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.48 Date Content Modified

Table 60. 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) on 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 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.49 Date Item Embargo Ends

Table 61. 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 the element is absent, the Item is not embargoed. If the element exists but is empty the end of the embargo is defined by the language in an **edNote** (page 136) element.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**DateTimeOrNullPropType** (page 274)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.50 Date Item First Created

Table 62. Date Item First Created

(XML) Data Model	Element
Namespace (prefix)	nar
Name	firstCreated
Title	Date Item First Created
Definition	A date plus a mandatory time with 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 275)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.51 Date Item Version Created

Table 63. Date Item Version Created

(XML) Data Model	Element
Namespace (prefix)	nar
Name	versionCreated
Title	Date Item Version Created
Definition	A date plus a mandatory time with 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 275)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.52 Date of Birth of Person

Table 64. Date of Birth

(XML) Data Model	Element
Namespace (prefix)	nar
Name	born
Title	Date of Birth of Person
Definition	The date (and optionally a time) on which a person was born.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.53 Date of Creation of Object

Table 65. Date of Creation of Object

(XML) Data Model	Element
Namespace (prefix)	nar
Name	created {Object}
Title	Date of Creation of Object
Definition	The date (and optionally a time) on which the object was created.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.54 Date of Creation of Point Of Interest

Table 66. Date of Creation of Point of Interest (POI)

(XML) Data Model	Element
Namespace (prefix)	nar
Name	created {POI}
Title	Date of Creation of Point Of Interest (POI)
Definition	The date (and optionally a time) on which the Point Of Interest was created.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.55 Date of Death of Person

Table 67. Date of Death of Person

(XML) Data Model	Element
Namespace (prefix)	nar
Name	died
Title	Date of Death of Person
Definition	The date (and optionally a time) on which the person died.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.56 Date of Dissolution of Geopolitical Area

Table 68. Date of Dissolution of Geopolitical Area

(XML) Data Model	Element
Namespace (prefix)	nar
Name	dissolved {geoArea}
Title	Date of Dissolution of Geopolitical Area
Definition	The date (and optionally a time) on which the geopolitical area was dissolved.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.57 Date of Dissolution of Organisation

Table 69. Date of Dissolution of Organisation

(XML) Data Model	Element
Namespace (prefix)	nar
Name	dissolved {Organisation}
Title	Date of Dissolution of Organisation
Definition	The date (and optionally a time) on which the organisation was dissolved.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.58 Date of Foundation of Geopolitical Area

Table 70. Date of Foundation of Geopolitical Area

(XML) Data Model	Element
Namespace (prefix)	nar
Name	founded {geoArea}
Title	Date of Foundation of Geopolitical Area
Definition	The date (and optionally a time) on which the geopolitical area was founded/established.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.59 Date of Foundation of Organisation

Table 71. Date of Foundation of Organisation

(XML) Data Model	Element
Namespace (prefix)	nar
Name	founded {Organisation}
Title	Date of Foundation of Organisation
Definition	The date (and optionally a time) on which the organisation founded/established.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.60 Date of Transmission

Table 72. Date of Transmission

(XML) Data Model	Element
Namespace (prefix)	nar
Name	sent
Title	Date of Transmission
Definition	A date plus a mandatory time with time zone of the 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	**DateTimePropType** (page 275)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.61 Dateline

Table 73. 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 292)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.62 Dates

Table 74. 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 252) (1)
	▪ **end** (page 139) (0..1) Or ▪ **duration** (page 135) (0..1)
	▪ **confirmation** (page 94) (0..1)
	▪ **Recurrence Group** (see **Table 75**) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.63 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 75. Recurrence Group Elements

Element Title	Element Name	Card	Described on Page
Recurrence Date	**rDate**	(0..unbounded)	226
Recurrence Rule	**rRule**	(0..unbounded)	227
Exclusion Date	**exDate**	(0..unbounded)	145
Exclusion Rule	**exRule**	(0..unbounded)	146

13.6.64 Date Resource Created

Table 76. Date Resource Created

(XML) Data Model	Element
Namespace (prefix)	nar
Name	created
Title	Date Resource Created
Definition	The date (and optionally the time with the time zone) on which the resource was created.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**TruncatedDateTimePropType** (page 302)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.65 Description

Table 77. 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: PCL		
Datatype	**BlockType** (page 269)		
Internally Ctrl Values			
Externally Ctrl Values	Recommended IPTC NewsCodes for the *role* attribute: http://cv.iptc.org/newscodes/descriptionrole/		
Attribute(s)	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.66 Deliverable Of

Table 78. Deliverable Of

(XML) Data Model	Element
Namespace (prefix)	nar
Name	deliverableOf
Title	Deliverable Of
Definition	A reference to the Planning Item and to one of its newsCoverage properties under which control this item has been published
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Link 1 Type** (page 293)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.67 Delivered Item Reference

Table 79. Delivered Item Reference

(XML) Data Model	Element
Namespace (prefix)	nar
Name	deliveredItemRef
Title	Delivered Item Reference
Definition	A reference to a G2 item which has been delivered pertaining to this newsCoverage.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Link 1 Type** (page 293)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.68 Delivery Information

Table 80. Delivery Information

(XML) Data Model	Element
Namespace (prefix)	nar
Name	delivery
Title	Delivery Information
Definition	A set of references to G2 items which have been delivered pertaining to this newsCoverage.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **deliveredItemRef** (page 132) (1..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.69 Destination

Table 81. 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)	If both are present the @literal and the property string value SHOULD be identical. If both are present but not identical @literal takes precedence		
XML Schema Spec	At: Both CCL and PCL		
Datatype	XML Schema string		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **qualifyAttributes** (page 309)	**Name**	**Datatype**
		qcode (0..1)	QCodeType
		literal (0..1)	XML Schema normalizedString
		type	QCodeType
		role	QCodeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.70 Duration

Table 82. 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 252) 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)	

13.6.71 Editorial Note

Table 83. 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. If edNote is a child element to plannedCoverage (EventsML-G2) this property provides additional natural language information about the planned coverage.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**BlockType** (page 269)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.72 Editorial Service

Table 84. 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. If service is a child element to plannedCoverage (EventsML-G2), this property indicates by which editorial service the planned G2 item(s) will be published.
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 298)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.73 Email Address

Table 85. 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 276)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.74 End Date/Time

Table 86. 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 267)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.75 Event

Table 87. 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 141) (1)		
	▪ **name** (page 91) (1..unbounded)		
	▪ **definition** (page 88) (0..unbounded)		
	▪ **facet** (page 147) (0..unbounded)		
	▪ **note** (page 201) (0..unbounded)		
	▪ **Concept Relationships Group** (page 263) (1)	**Element Name**	**Page**
		broader (0..unbounded)	79
		narrower (0..unbounded)	194
		related (0..unbounded)	229
		sameAs {Relationship} (0..unbounded)	244
XML Schema Note(s)			
Example(s)			

13.6.76 Event Details

Table 88. 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 128) (1)
	▪ **occurStatus** (page 203) (0..1)
	▪ **newsCoverageStatus** (page 198) (0..1)
	▪ **registration** (page 228) (1..unbounded)
	▪ **accessStatus** (page 65) (0..unbounded)
	▪ **subject** (page 253) (0..unbounded);
	▪ **location** (page 142) (0..unbounded)
	▪ **participant** (page 209) (0..unbounded)
	▪ **participationRequirement** (page 210) (0..unbounded)
	▪ **organiser** (page 207) (0..unbounded)
	▪ **contactInfo** (page 95) (0..unbounded)
	▪ **language** (page 185) (0..unbounded)
	▪ **newsCoverage {Concept}** (page 195) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.77 Event Location

Table 89. 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 281)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeType** (page 297); A refinement on the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.78 Events Wrapper

Table 90. 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 140) (1..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.79 Excluded Audience

Table 91. 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 268)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.80 Exclusion Date

Table 92. 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 272)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.81 Exclusion Rule

Table 93. 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 299)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.82 Facet (DEPRECATED)

Table 94. Facet

(XML) Data Model	Element
Namespace (prefix)	nar
Name	facet
Title	Facet (DEPRECATED)
Definition	**In NAR 1.8 (EventsML-G2 1.6, NewsML-G2 2.7) and later, facet is deprecated and SHOULD NOT (see RFC 2119) be used**, the "related" property should be used instead (its definition was: An intrinsic property of the concept.)
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**FlexPropType** (page 287)
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 297); 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.
	▪ **timeValidityAttributes** (page 307) — **Name**: validfrom (0..1), **Datatype**: DateOptTimeType; validto (0..1), DateOptTimeType
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.83 Fax Number

Table 95. 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 276)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.84 File Name

Table 96. 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 306)	**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)			

13.6.85 G2 Content Type

Table 97. G2 Content Type

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	g2ContentType		
Title	G2 Content Type		
Definition	The kind of planned G2 item(s).		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	XML Schema String		
Internally Ctrl Values	Any of the G2-Standards specific IANA MIME (see **MIME Types** on page 61) types like application/vnd.iptc.g2.*item+xml. See: http://www.iana.org/assignments/media-types/application/		
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 306)	**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)			

13.6.86 G2 Item Class

Table 98. G2 Item Class

(XML) Data Model	Element
Namespace (prefix)	nar
Name	itemClass
Title	G2 Item Class
Definition	The nature of the planned G2 item(s).
User Note(s)	MUST correspond to the itemClass property of the planned item.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**QualPropType** (page 298)
Internally Ctrl Values	Any of the G2-Standards specific IANA MIME (see **MIME Types** on page 61) types such as application/vnd.iptc.g2.*item+xml. See: http://www.iana.org/assignments/media-types/application/
Externally Ctrl Values	Recommended IPTC NewsCodes: http://cv.iptc.org/newscodes/ninature/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.87 Generator Tool

Table 99. 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)	Where a role IS NOT specified, the Generator Tool applies to the most recent item generation stage. Where a role IS specified, the Generator Tool applies to the item generation stage identified by the role.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Extends **VersionedStringType** (page 305)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeType** (page 297); Identifies the stage at which this generator was used.
Child Element(s)	
XML Schema Note(s)	

Example(s)	`<!-- Generator: implicit -->`
	`<generator versioninfo="00.00.01">G3:IIM:FH</generator>`
	`<!-- Generator: explicit, by role -->`
	`<generator versioninfo="1.22.109" role="gen-Role:MDN">Janus</generator>`

13.6.88 Genre

Table 100. 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 278)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.89 Geographic Position

Table 101. 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)	

13.6.90 Geopolitical Area Details

Table 102. 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 154) (0..1)
	▪ **founded {geoArea}** (page 124) (0..1)
	▪ **dissolved {geoArea}** (page 122) (0..1)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.91 Group

Table 103. 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 297); The part this group plays within its container.
	▪ mode (0..unbounded); **QCodeType** (page 297); 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.
	▪ **i18nAttributes** (page 306) — **Name**: xml:lang (0..1); **Datatype**: XML Schema language **Name**: dir (0..1); **Datatype**: XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **groupRef** (page 157) (0..unbounded)
	▪ **itemRef** (page 180) (0..unbounded)
	▪ **conceptRef** (page 92) (0..unbounded)
XML Schema Note(s)	The local identifier (id) common to all elements at PCL provides a local identifier for groups.
Example(s)	

13.6.92 Group Reference

Table 104. 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)	

13.6.93 Group Set

Table 105. 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 156) (1..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.94 Hash Value

Table 106. Hash Value

(XML) Data Model	Element
Namespace (prefix)	nar
Name	hash
Title	Hash Value
Definition	A hash value of parts of an item as defined by the hashscope attribute
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	XML Schema string
Internally Ctrl Values	
Externally Ctrl Values	Recommended IPTC NewsCodes CVs: for @hashtype: http://cv.iptc.org/newscodes/hashtype/ for @hashscope: http://cv.iptc.org/newscodes/hashscope/
Attribute(s)	▪ **editAttributes** (page 306) **Name** / **Datatype** id (0..1) / XML Schema ID creator (0..1) / QCodeType modified (0..1) / DateOptTimeType ▪ hashtype (1..1); **QCodeType** (page 297); The hash algorithm used for creating the hash value ▪ scope (0..1); **QCodeType** (page 297); The scope of a G2 item's content which is the reference for creating the hash value. If the attribute is omitted http://cv.iptc.org/newscodes/hashscope/content is the default value
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.95 Headline

Table 107. 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 292)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.96 Hierarchy Info

Table 108. Hierarchy Info

(XML) Data Model	Element
Namespace (prefix)	nar
Name	hierarchyInfo
Title	Hierarchy Info
Definition	Indicates the nature of the Item.
User Note(s)	Represents the position of a concept in a hierarchical taxonomy tree by a sequence of QCode tokens representing the ancestor concepts and this concept.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	XML Schema NMTOKENS
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	From the Media Topic NewsCodes (alias="mtp") using assumed codes: The concept "adoption" has QCode mtp:2788 Its parent is the concept "family" with the QCode mtp:2780 The parent of "family" is the top level concept "society" with the Qcode mtp:1400 The resulting Hierarchy Info value is <hierarchyInfo>mtp:1400 mtp:2780 mtp:2788</hierarchyInfo>

13.6.97 Hop

Table 109. Hop in Hop History

(XML) Data Model	Element
Namespace (prefix)	nar
Name	hop
Title	Hop in Hop History
Definition	A single hop of the Hop History. The details of the hop entry should reflect the actions taken by a party.
User Note(s)	The timestamp of the hop element reflects the time of forwarding the object while the timestamp of an action reflects the time of performing that individual action.
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ seq (0..1); XML Schema nonNegativeInteger; The sequential value of this Hop in a sequence of Hops of a Hop History. Values need not to be consecutive. The sequence starts with the lowest value.
	▪ timestamp (0..1); XML Schema DateTime; The date and optionally the time (with a time zone) when this item's content was forwarded.
Child Element(s)	▪ **party** (page 213) (0..1)
	▪ **action** (page 67) (0..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.98 Hop History

Table 110. Hop History

(XML) Data Model	Element
Namespace (prefix)	nar
Name	hopHistory
Title	Hop History
Definition	A history of the creation and modifications of the content object of this item, expressed as a sequence of hops.
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)	▪ **hop** (page 162) (1..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.99 Icon

Table 111. Icon

(XML) Data Model	Element
Namespace (prefix)	nar
Name	icon
Title	Icon
Definition	An iconic visual representation of the content.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ rendition (0..1); **QCodeType** (page 297); Identifies the rendition of the target resource. If the target resource is an item providing multiple renditions then this attribute is used to identify the rendition to be used.
	▪ width (0..1); xsd:nonNegativeInteger; The width of of visual content.
	▪ widthunit (0..1); **QCodeType** (page 297); If present it defines the width unit for the width.
	▪ height (0..1); xsd:nonNegativeInteger; The height of visual content.
	▪ heightunit (0..1); **QCodeType** (page 297); If present it defines the height unit for the height.

	Name	Datatype
▪ **targetResourceAttributes** (page 311)	href (0..1)	IRIType
	residref (0..1)	XML Schema string
	version (0..1)	XML Schema positiveInteger
	contenttype (0..1)	XML Schema string
	format (0..1)	QCodeType
	size (0..1)	XML Schema non NegativeInteger
▪ **editAttributes** (page 306)	**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)	

13.6.100 Inline Concept Marker

Table 112. 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 297); 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 297); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **quantifyAttributes** (page 309)	**Name**	**Datatype**
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
		why (0..1)	QCodeType
Child Element(s)	▪ **span** (page 251) (0..unbounded)		
	▪ **ruby** (page 240) (0..unbounded)		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.6.101 Inline Data

Table 113. Inline Data

(XML) Data Model	Element
Namespace (prefix)	nar
Name	inlineData
Title	Inline Data
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 113. Inline Data

Attribute(s)	▪ encoding (0..1); **QCodeType** (page 297); Specifies the encoding applied to the content before inclusion in the content.		
	▪ contenttype (0..1); XML Schema string; The IANA (Internet Assigned Numbers Authority) MIME type of the target resource.		
	▪ format (0..1); QCodeType; Refinement of a generic content type (i.e. IANA MIME type).		
	▪ **newsContentAttributes** (page 310)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		rendition (0..1)	QCodeType
		generator (0..1)	XML Schema string
		generated (0..1)	DateOptTimeType
	▪ **newsContentCharacteristics** (page 311) (all: 0..1)	**Name**	**Datatype**
		wordcount	XML Schema nonNegativeInteger
		width	XML Schema nonNegativeInteger
		widthunit	QCodeType
		height	XML Schema nonNegativeInteger
		heightunit	QCodeType
		orientation	XML Schema nonNegativeInteger
		colourspace	QCodeType
		colourindicator	QCodeType
		resolution	XML Schema positiveInteger
		duration	XML Schema nonNegativeInteger
		durationunit	QCodeType
		audiocodec	QCodeType
		audiobitrate	XML Schema positiveInteger
		audiovbr	XML Schema boolean
		audiosamplesize	XML Schema positiveInteger
		audiosamplerate	XML Schema positiveInteger
		audiochannels	QCodeType
		videocodec	QCodeType
		videoavgbitrate	XML Schema positiveInteger
		videovbr	XML Schema boolean
		videoframerate	XML Schema decimal
		videoscan	enumeration progressive/interlaced
		videoaspectratio	XML Schema nomalizedString
		videosampling	XML Schema nomalizedString
		videoscaling	QCodeType
		videodefinition	QCodeType
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)			
XML Schema Note(s)			

Table 113. Inline Data

Example(s)	

13.6.102 Inline Reference

Table 114. 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 280)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ idrefs (0..1); XML Schema IDREFS; A set of local identifiers of inline content.		
	▪ **quantifyAttributes** (page 309)	**Name**	**Datatype**
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
		why (0..1)	QCodeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.103 Inline XML

Table 115. Inline XML

(XML) Data Model	Element
Namespace (prefix)	nar
Name	inlineXML
Title	Inline XML
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	

Table 115. Inline XML

Attrtibute(s)	▪ contenttype (0..1); XML Schema string; The IANA (Internet Assigned Numbers Authority) MIME type of the target resource.		
	▪ format (0..1); QCodeType; Refinement of a generic content type (i.e. IANA MIME type).		
	▪ **newsContentAttributes** (page 310)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		rendition (0..1)	QCodeType
		generator (0..1)	XML Schema string
		generated (0..1)	DateOptTimeType
	▪ **newsContentCharacteristics** (page 311) (all: 0..1)	**Name**	**Datatype**
		wordcount	XML Schema nonNegativeInteger
		width	XML Schema nonNegativeInteger
		widthunit	QCodeType
		height	XML Schema nonNegativeInteger
		heightunit	QCodeType
		orientation	XML Schema nonNegativeInteger
		colourspace	QCodeType
		colourindicator	QCodeType
		resolution	XML Schema positiveInteger
		duration	XML Schema nonNegativeInteger
		durationunit	QCodeType
		audiocodec	QCodeType
		audiobitrate	XML Schema positiveInteger
		audiovbr	XML Schema boolean
		audiosamplesize	XML Schema positiveInteger
		audiosamplerate	XML Schema positiveInteger
		audiochannels	QCodeType
		videocodec	QCodeType
		videoavgbitrate	XML Schema positiveInteger
		videovbr	XML Schema boolean
		videoframerate	XML Schema decimal
		videoscan	enumeration progressive/interlaced
		videoaspectratio	XML Schema nomalizedString
		videosampling	XML Schema nomalizedString
		videoscaling	QCodeType
		videodefinition	QCodeType
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Table 115. Inline XML

Child Element(s)	▪ Plug-in Point (0..1). XML content from any namespace.
XML Schema Note(s)	
Example(s)	

13.6.104 Instance Of

Table 116. 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 280)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.105 Instant Messaging Address

Table 117. 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 277)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.106 Information Source

Table 118. 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 279)
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)	

13.6.107 Item Class

Table 119. 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 298)
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)	

13.6.108 Item Count

Table 120. Item Count

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	itemCount		
Title	Item Count		
Definition	The number of planned G2 items of the kind indicated by the context and expressed by a range.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ rangefrom (1); XML Schema nonNegativeInteger; The lower limit of the range of planned items		
	▪ rangeto (1); XML Schema positiveInteger; The upper limit of the range of planned items		
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.109 Item Metadata

Table 121. 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 265) (1)
	▪ **link** (page 187) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.

Element Name	Page
provider (1)	102
embargoed (0..1)	115
firstCreated (0..1)	116
versionCreated (1)	117
edNote (0..unbounded)	136
service (0..unbounded)	137
filename (0..1)	149
itemClass (1)	176
pubStatus (0..1)	225
role (0..1)	239
title (0..unbounded)	182
altRep (0..unbounded)	72
generator (0..1)	152
instanceOf (0..unbounded)	173
memberOf (0..unbounded)	192
profile (0..1)	223
signal (0..unbounded)	249
deliverableOf (0..1)	131
deliverableOf (0..unbounded)	131

Table 121. Item Metadata (Continued)

XML Schema Note(s)	

Example(s)	Add a Hash Value for the Inline XML content of a News Item, using a provider-specific mix of content and metadata fields to generate the hash. (Scope explicitly defined)
	`<itemMeta>`
	`...`
	`<hash hashtype="htype:MD5" scope="hscope:provmix">hash-value.....`
	`</hash>`
	`</itemMeta>`

13.6.110 Item Reference

Table 122. 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 293)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.111 Item Set

Table 123. 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 199) (1..unbounded)
	▪ **conceptItem** (page 90) (1..unbounded)
	▪ **packageItem** (page 208) (1..unbounded)
	▪ **knowledgeItem** (page 184) (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)	

13.6.112 Item Title

Table 124. 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 292)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.113 Keyword

Table 125. Keyword

(XML) Data Model	Element
Namespace (prefix)	nar
Name	keyword
Title	Keyword
Definition	Free-text term to be used for indexing or finding the content by text-based search engines.
User Note(s)	This property may be used in parallel with other properties to describe content like subject or genre which use QCodes or literals to identify an assigned concept. Providers should define if and how the values of keyword properties contained in their items complement, or overlap with, the values of other properties such as subject or genre.
Implementation Note(s)	Be aware of the lexical space restrictions for an XML Schema normalizedString - see XML Schema specifications.
XML Schema Spec	At: PCL
Datatype	Extends **IntlStringType** (page 290)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 295); A refinement of the semantics of the keyword.

	Name	Datatype
▪ **quantifyAttributes** (page 309)	confidence (0..1)	Int100Type
	relevance (0..1)	Int100Type
	why (0..1)	QCodeType
▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
	rank (0..1)	XML Schema nonNegativeIntege

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

13.6.114 Knowledge Item

Table 126. 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 {Knowledge}** (page 97) (0..1)
	▪ **assert** (page 74) (0..unbounded)
	▪ **inlineRef** (page 169) (0..unbounded)
	▪ **conceptSet** (page 93) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.115 Language

Table 127. 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 be valid BCP 47 language tags. 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 295); A refinement of the semantics of the property.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)	▪ **name** (page 224) (0..unbounded)		
XML Schema Note(s)			
Example(s)			

13.6.116 Line Break

Table 128. 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)	

13.6.117 Link

Table 129. 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 293)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.118 Locality

Table 130. 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 280)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.119 Located

Table 131. 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 281)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.120 Location

Table 132. 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 281)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 307)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.121 Location Details

Table 133. 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 269)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.122 Member Of

Table 134. 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 280)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.123 Message Header

Table 135. 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)	If any QCode is used within the News Message header then a catalog and/or a catalogRef element MUST be included in the header. The scope of the scheme elements of the local and/or remote catalog(s) is limited to the header element and its descendants.
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **sent** (page 126) (1); May not be updated in case of a message retransmission.
	▪ **sender** (page 248) (0..1); The structure of this string is not specified by the IPTC. Best practice is to identify a sender by its domain name.
	▪ **catalogRef** (page 232) (0..unbounded)
	▪ **catalog** (page 82) (0..unbounded)
	▪ **transmitId** (page 257) (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 222) (0..1)
	▪ **origin** (page 205) (0..1); This string structure is not specified by the IPTC.
	▪ **destination** (page 134) (0..unbounded)
	▪ **channel** (page 85) (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 256) (0..unbounded)
	▪ **signal** (page 249) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.124 Narrower

Table 136. 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 300)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.125 News Coverage (Concept Item)

Table 137. News Coverage for a Concept Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	newsCoverage {Concept}		
Title	News Coverage for a Concept Item (LEGACY)		
Definition	Structured and 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)	Be aware that in EventsML-G2 version 1.6 this element was classified as LEGACY. From that version on a standalone Planning Item is available to hold an even extended set of information about planned coverage. Its major advantage is that coverage can be planned without having to update - and version - concept items for event concepts.		
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ role (0..1); **QCodeType** (page 297); Refines the semantics of the property.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)	▪ **g2ContentType** (page 150) (0..1)		
	▪ **itemClass** (page 151) (0..1)		
	▪ **assignedTo** (page 75) (0..1)		
	▪ **scheduled** (page 246) (0..1)		
	▪ **service** (page 137) (0..unbounded)		
	▪ **edNote** (page 136) (0..unbounded)		
	▪ **Descriptive Metadata Group** (page 265) (0..1)	**Element Name**	**Page**
		by (0..unbounded)	80
		creditline (0..unbounded)	110
		dateline (0..unbounded)	127
		description (0..unbounded)	130
		genre (0..unbounded)	153
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
XML Schema Note(s)			
Example(s)			

13.6.126 News Coverage (Planning Item)

Table 138. News Coverage for a Planning Item

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	newsCoverage {Planning}		
Title	News Coverage for a Planning Item		
Definition	Information about the planned and delivered news coverage of the news provider.		
User Note(s)	A new newsCoverage property must be created for each set of planning details which contains different values. Different would be typically the g2contentType and/or the itemClass; or one or more of the descriptive metadata properties for the planned items.		
Implementation Note(s)			
XML Schema Spec	At: Both CCL and PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **Persistent Editing Attributes** (page 307)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)	▪ **planning** (page 216) (1)		
	▪ **delivery** (page 133) (0..1)		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.6.127 News Coverage Set

Table 139. News Coverage Set

(XML) Data Model	Element
Namespace (prefix)	nar
Name	newsCoverageSet
Title	News Coverage Set
Definition	A set of data about planned and delivered news coverage. This information is aimed at the editorial staff of the receiver.
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)	▪ **newsCoverage {Planning}** (page 196) (1..unbounded)
XML Schema Note(s)	
Example(s)	

13.6.128 News Coverage Status

Table 140. News Coverage Status

(XML) Data Model	Element
Namespace (prefix)	nar
Name	newsCoverageStatus
Title	News Coverage Status
Definition	Indicates the certainty of the news coverage of the event
User Note(s)	Indicating a decision of coverage: If a specific coverage was agreed by the news provider the newsCoverageStatus has to be set to code “int” (coverage intended) and at least one newsCoverage element with coverage details MUST be added to the eventDetails.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**QualPropType** (page 298)
Internally Ctrl Values	
Externally Ctrl Values	Highly recommended IPTC NewsCodes: http://cv.iptc.org/newscodes/newscoveragestatus/
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.129 News Item

Table 141. News Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	newsItem
Title	News Item
Definition	An Item containing news-related information.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**AnyItemType** (page 266)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta {News}** (page 98) (0..1)
	▪ **partMeta** (page 211) (0..1)
	▪ **assert** (page 74) (0..unbounded)
	▪ **inlineRef** (page 169) (0..unbounded)
	▪ **contentSet** (page 103) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.130 News Message

Table 142. 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 193) (1)
	▪ **itemSet** (page 181) (1)
XML Schema Note(s)	
Example(s)	

13.6.131 Note

Table 143. 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 269)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 307)	**Name**	**Datatype**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.132 Object Details

Table 144. Object Details

(XML) Data Model	Element
Namespace (prefix)	nar
Name	objectDetails
Title	Object Details
Definition	A set of properties representing an object.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **created {Object}** (page 119) (0..1)
	▪ **creator** (page 109) (0..unbounded)
	▪ **copyrightNotice** (page 106) (0..unbounded)
	▪ **ceasedToExist {Object}** (page 111) (0..1)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.133 Occurrence Status

Table 145. 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 296)
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)	

13.6.134 Opening Hours

Table 146. 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 292)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.135 Origin

Table 147. 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)	If both are present the @literal and the property string value SHOULD be identical. If both are present but not identical @literal takes precedence		
XML Schema Spec	At: Both CCL and PCL		
Datatype	XML Schema string		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **qualifyAttributes** (page 309)	**Name**	**Datatype**
		qcode (0..1)	QCodeType
		literal (0..1)	XML Schema normalizedString
		type	QCodeType
		role	QCodeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.136 Organisation Details

Table 148. 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 {Organisation}** (page 125) (0..1)
	▪ **dissolved {Organisation}** (page 123) (0..1)
	▪ **location** (page 190) (0..unbounded)
	▪ **contactInfo** (page 95) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.137 Organiser

Table 149. 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 279)
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)	

13.6.138 Package Item

Table 150. 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 266)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta {Package}** (page 99) (0..1)
	▪ **assert** (page 74) (0..unbounded)
	▪ **inlineRef** (page 169) (0..unbounded)
	▪ **groupSet** (page 158) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.139 Participant

Table 151. 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 279)
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)	

13.6.140 Participation Requirement

Table 152. 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 280)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeType** (page 297); Refines the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.141 Part of Content Metadata

Table 153. Part of Content Metadata

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	partMeta		
Title	Part of Content Metadata		
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.		
	▪ seq (0..1); XML Schema nonNegativeInteger; The sequence number of the part.		
	▪ contentrefs (0..1); XML Schema IDREFS; A list of identifiers of XML elements containing content which is described by this partMeta structure.		
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Table 153. Part of Content Metadata (Continued)

Child Element(s)	▪ **role** (page 238) (0..1)		
	▪ **icon** (page 164) (0..unbounded); If multiple icon elements are present within a single contentMeta or partMeta property they MUST represent the same visual content, only differentiated by rendition, contentType or format.		
	▪ **timeDelim** (page 254) (0..1)		
	▪ **regionDelim** (page 261) (0..1)		
	▪ **Administrative Metadata Group** (page 264)	**Element Name**	**Page**
		urgency (0..1)	259
		contentCreated (0..1)	113
		contentModified (0..1)	114
		located (0..unbounded)	189
		infoSource (0..unbounded)	175
		creator (0..unbounded)	109
		contributor (0..unbounded)	104
		audience (0..unbounded)	76
		exclAudience (0..unbounded)	144
		altId (0..unbounded)	70
	▪ **Descriptive Metadata Group** (page 265) (0..1)	**Element Name**	**Page**
		by (0..unbounded)	80
		creditline (0..unbounded)	110
		dateline (0..unbounded)	127
		description (0..unbounded)	130
		genre (0..unbounded)	153
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			

Example(s)	`<!-- Example: Defining a 1.5 second part running from 2.0S to 3.5S, using 'normalPlayTime', qualified as a 'sting' -->`
	`<partMeta>`
	`<role qcode="partRole:string"/>`
	`<timeDelim start="00:00:02.000" end="00:00:03.500" time-unit="timeunit:normalPlayTime"/>`
	`</partMeta>`		

13.6.142 Party (Hopy History)

Table 154. Party of the Hop History

(XML) Data Model	Element
Namespace (prefix)	nar
Name	party
Title	Party of the Hop History
Definition	A party involved in this hop of the Hop History
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Flex1PartyPropType** (page 279)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.143 Person Details

Table 155. 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 118) (0..1)
	▪ **died** (page 121) (0..1)
	▪ **affiliation** (page 69) (0..unbounded)
	▪ **contactInfo** (page 95) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.144 Phone Number

Table 156. 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 277)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.145 Planning Details

Table 157. Planning Details

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	planning		
Title	Planning Details		
Definition	Details about the planned news coverage		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype			
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)			
Child Element(s)	▪ **g2ContentType** (page 150) (0..1)		
	▪ **itemClass** (page 151) (0..1)		
	▪ **itemCount** (page 177) (0..1)		
	▪ **assignedTo** (page 75) (0..1)		
	▪ **scheduled** (page 246) (0..1)		
	▪ **service** (page 137) (0..unbounded)		
	▪ **edNote** (page 136) (0..unbounded)		
	▪ **Descriptive Metadata Group** (page 265) (0..1)	**Element Name**	**Page**
		by (0..unbounded)	80
		creditline (0..unbounded)	110
		dateline (0..unbounded)	127
		description (0..unbounded)	130
		genre (0..unbounded)	153
		headline (0..unbounded)	160
		keyword (0..unbounded)	183
		language (0..unbounded)	185
		slugline (0..unbounded)	250
		subject (0..unbounded)	253
XML Schema Note(s)			
Example(s)			

13.6.146 Planning Item

Table 158. Planning Item

(XML) Data Model	Element
Namespace (prefix)	nar
Name	planningItem
Title	Planning Item
Definition	An Item containing information about the planning and delivery of news coverage.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**AnyItemType** (page 266)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **contentMeta {Planning}** (page 101) (0..1)
	▪ **assert** (page 74) (0..unbounded)
	▪ **inlineRef** (page 169) (0..unbounded)
	▪ **newsCoverageSet** (page 197) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.147 POI Details

Table 159. 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: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **position** (page 154) (0..1)
	▪ **adress** (page 220) (0..1)
	▪ **openHours** (page 204) (0..1)
	▪ **capacity** (page 81) (0..1)
	▪ **access** (page 64) (0..unbounded)
	▪ **details** (page 191) (0..unbounded)
	▪ **contactInfo** (page 95) (0..unbounded)
	▪ **created {POI}** (page 120) (0..1)
	▪ **ceasedToExist {POI}** (page 112) (0..1)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.148 Postal Address

Table 160. 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 295); A refinement of the semantics of the postal address.
Child Element(s)	▪ **line** (page 68) (0..unbounded)
	▪ **locality** (page 188) (0..1)
	▪ **area** (page 108) (0..1)
	▪ **country** (page 107) (0..1)
	▪ **postalCode** (page 221) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.149 Postal Address of a Point of Interest

Table 161. Postal Adress of A Point of Interest

(XML) Data Model	Element
Namespace (prefix)	nar
Name	adress
Title	Postal Adress of A Point of Interest
Definition	A postal address for the location of a Point Of Interest.
User Note(s)	This address may be different from an address required to contact the Point Of Interest or the organisation running or maintaining it, that address is provided under a contactInfo element.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	▪ **line** (page 68) (0..unbounded)
	▪ **locality** (page 188) (0..1)
	▪ **area** (page 108) (0..1)
	▪ **country** (page 107) (0..1)
	▪ **postalCode** (page 221) (0..1)
XML Schema Note(s)	
Example(s)	

13.6.150 Postal Code

Table 162. 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 290)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.151 Priority

Table 163. 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 289)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.152 Profile

Table 164. 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 305)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.153 Property Value Name

Table 165. 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 290)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.154 Publish Status

Table 166. 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 298)
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)	

13.6.155 Recurrence Date

Table 167. 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 272)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.156 Recurrence Rule

Table 168. 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 299)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.157 Registration

Table 169. 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 269)
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)	

13.6.158 Related Concept

Table 170. Related Concept

(XML) Data Model	Element
Namespace (prefix)	nar
Name	related
Title	Related Concept
Definition	A related concept, where the relationship is different from sameAs, broader, or narrower.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	Extends **RelatedConceptType** (page 300)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ value; (0..1); XML Schema String; The related property's value (see also: qcode and literal attributes and Note 2 below the table.)
	▪ valuedatatype; (0..1); XML Schema QName; The datatype of the value attribute - it MUST be one of the built-in datatypes defined by a W3C XML Schema specification.
	▪ valueunit; (0..1); **QCode Type** (page 297); The unit of the value attribute.
Child Element(s)	▪ **bag** (page 77) (0..1)

Table 170. Related Concept

XML Schema Note(s)	

Table 170. Related Concept

Note 1) On the RDF which applies to the related property:
As pointed out in the basics about the G2-Standards the properties of an item represent an RDF triple with an RDF Predicate which is defined by the semantics of the property. The related property is slightly different as its RDF Predicate is defined by the semantics of the rel attribute:

Subject	Predicate	Object
this concept	@rel value	@qcode OR @literal OR @value

Note 2) On how to express the value of the related property:
One out of three alternatives to express the value of the related property can be chosen, each alternative uses only a single attribute for expressing the value:

	Attribute **qcode**	Attribute **literal**	Attribute **value**	Applicable use case
1	present	must not be present	must not be present	The value is a concept from a controlled vocabulary
2	must not be present	present	must not be present	The value is a concept which is not from a controlled vocabulary
3	must not be present	must not be present	present	The value is not a concept.

In a more formal way: the use of the attributes qcode, literal and value is mutually exclusive.

Note 3) on the use of the attributes value, valuedatatype and valueunit:
Alternatives to a concept as the value of the related property can be expressed by using these attributes:
A value MUST be provided.
A valuedatatype MUST be provided.
A valueunit MAY be provided, depending on how the value is measured: for any quantity having a unit the used unit must be indicated by this valueunit attribute. For any value not using a unit the valueunit attribute is omitted.

Note(s) & Example(s)

Example 1 for Values:
Expressing a recommendation from an analyst:

```
<concept xmlns:xs="http://www.w3.org/2001/XMLSchema">
...
  <related rel="crel:price_new" value="44" valueunit="iso4217:EUR "
valuedatatype="xs:decimal" />
  <related rel="crel:price_old" value="39" valueunit="iso4217:EUR "
valuedatatype="xs:decimal" />
  <related rel="crel:rank_old" value="Hold" value-
unit="valunits:trRanks" valuedatatype="xs:string" />
  <related rel="crel:rank_new" value="Buy" value-
unit="valunits:trRanks" valuedatatype="xs:string" />
  <related rel="crel:rank_new" qcode="trRanks:Buy" />
...
</concept>
```

Example 2 for Values:
Expressing some details about a sports game:

```
<concept xmlns:xs="http://www.w3.org/2001/XMLSchema">
...
  <related rel="crel:TopLeagueRanking" value="3" valueunit="
valunits:ranks" valuedatatype="xs:nonNegativeInteger"  />
  <related rel="crel:goalsShot" value="4" valueunit=" valunits:score-
count" valuedatatype="xs:nonNegativeInteger"  />
  <related rel="crel:goalScorer" qcode="playerList:Pl4799" />
  <related rel="crel:goalScorer" qcode="playerList:Pl9832" />
...
</concept>
```

13.6.159 Remote Catalog Reference

Table 171. 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 291); A hyperlink to a remote catalog.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)	▪ **title** (page 182) (0..unbounded)		
XML Schema Note(s)			
Example(s)			

13.6.160 Remote Content

Table 172. Remote Content

(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)	To identify the remote resource either the residref attribute or the href attribute MUST be set, optionally both MAY be used in parallel. The residref attribute identifies a managed remote resource by its globally unique identifier (if the resource has such an identifier), while the href attribute identifies the location of the remote resource in e.g. a (remote) file system. If the remote resource is managed - like an item - and consequently the residref attribute is used, a version attribute MAY indicate the resource's version; in the absence of version information, the remote resource is the latest version available. The Width Unit and Height Unit may take the following values, taken from an IPTC defined controlled vocabulary: lines, pixels, points (more units are defined by this CV, check the most recent version).
Implementation Note(s)	If the Width Unit and/or Height Unit IS NOT present, the default value(s) in **Table 254** on page 313 MUST be assumed
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	Mandatory IPTC NewsCodes: http://cv.iptc.org/newscodes/dimensionunit/

Table 172. Remote Content (Continued)

Attribute(s)	▪ **newsContentAttributes** (page 310)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		rendition (0..1)	QCodeType
		generator (0..1)	XML Schema string
		generated (0..1)	DateOptTimeType
	▪ **targetResourceAttributes** (page 311)	**Name**	**Datatype**
		href (0..1)	IRIType
		residref (0..1)	XML Schema string
		version (0..1)	XML Schema positiveInteger
		contenttype (0..1)	XML Schema string
		format (0..1)	QCodeType
		size (0..1)	XML Schema non NegativeInteger
	▪ **timeValidityAttributes** (page 307)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
	▪ **newsContentCharacteristics** (page 311) (all: 0..1)	**Name**	**Datatype**
		wordcount	nonNegativeInteger
		width	XML Schema nonNegativeInteger
		widthunit	QCodeType
		height	XML Schema nonNegativeInteger
		heightunit	QCodeType
		orientation	XML S nonNegativeInteger
		colourspace	QCodeType
		colourindicator	QCodeType
		resolution	positiveInteger
		duration	XML S nonNegativeInteger
		durationunit	QCodeType
		audiocodec	QCodeType
		audiobitrate	XML Schema positiveInteger
		audiovbr	XML Schema boolean
		audiosamplesize	XML Schema positiveInteger
		audiosamplerate	XML Schema positiveInteger
		audiochannels	QCodeType
		videocodec	QCodeType
		videoavgbitrate	XML Schema positiveInteger
		videovbr	XML Schema boolean
		videoframerate	XML Schema decimal
		videoscan	enum progressive/interlaced
		videoaspectratio	XML Schema nomalizedString
		videosampling	XML Schema nomalizedString
		videoscaling	QCodeType
		videodefinition	QCodeType
	▪ language (0..1); XML Schema language; The language used by the remote content.		

Table 172. Remote Content (Continued)

Child Element(s)	▪ **altLoc** (page 71) (0..unbounded)
	▪ **altId** (page 70) (0..unbounded)
	▪ **channel** (page 85) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	

Example(s)	`<!-- RC: Picture, using implicit default dimensionunit:pixels -->`
	`<remoteContent residref="tag:reuters.com,0000:binary_BTRE4A31LE800-THUMBNAIL" rendition="rend:thumbnail" contenttype="image/jpeg" format="fmt:jpegBaseline" width="100" height="100" />`
	`<!-- RC: Graphic, using explicit dimensionunits -->`
	`<remoteContent residref="tag:reuters.com,0000:binary_BTRE37913MM00-THUMBNAIL" rendition="rend:thumbnail" contenttype="image/gif" format="fmt:gif87a" width="100" widthunit="dimensionunit:points" height="100" heightunit="dimensionunit:points"/>`

13.6.161 Remote Information about a Concept

Table 173. Remote Information about a Concept

(XML) Data Model	Element
Namespace (prefix)	nar
Name	remoteInfo
Title	Remote Information about a Concept
Definition	Link to an item or a web resource which provides information about the concept.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**Link1Type** (page 293)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.162 Rights Information

Table 174. Rights Information

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rightsInfo
Title	Rights Information
Definition	A set of properties representing rights associated with the entire Item oar parts of 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; Reference(s) to the part(s) of an Item to which the rightsInfo element applies. When referencing part(s) of the content of an Item, idrefs must include the partid value of a **partMeta** (page 211) element which in turn references the part of the content.
	▪ scope; (0..1), QCodeListType; Indicates to which part(s) of an Item the rightsInfo element applies. If the attribute does not exist then rightsInfo applies to all parts of the Item. Mandatory NewsCodes scheme for the values: http://cv.iptc.org/newscodes/riscope/.
	▪ aspect; (0..1), QCodeListType; Indicates to which rights-related aspect(s) of an Item or part(s) of an Item the rightsInfo element applies. If the attribute does not exist then rightsInfo applies to all aspects. Mandatory NewsCodes scheme for the values: http://cv.iptc.org/newscodes/raspect/.
	▪ **timeValidityAttributes** (page 307) — **Name**: validfrom (0..1), validto (0..1); **DataType**: DateOptTimeType, DateOptTimeType
Child Element(s)	▪ **accountable** (page 66) (0..1)
	▪ **copyrightHolder** (page 105) (0..1)
	▪ **copyrightNotice** (page 106) (0..unbounded)
	▪ **usageTerms** (page 260) (0..unbounded)
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.
XML Schema Note(s)	
Example(s)	

13.6.163 Role in the Content Stream

Table 175. Role in Content Stream

(XML) Data Model	Element
Namespace (prefix)	nar
Name	role
Title	Role in Content Stream
Definition	The role in the overall content stream.
User Note(s)	This property may indicate the role of the content part in a piece of streaming media. Examples (video): “sting”, “slate”, etc.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**QualPropType** (page 298)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.164 Role in the Workflow

Table 176. 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 298)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.165 Ruby

Table 177. 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 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **rb** (page 241) (1)		
	▪ **rp** (page 242) (see XML Schema note below)		
	▪ **rt** (page 243) (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>`		

13.6.166 Ruby Base

Table 178. 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 240).
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)	

13.6.167 Ruby Parenthesis

Table 179. 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 240).
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>`

13.6.168 Ruby Text

Table 180. Ruby Text

(XML) Data Model	Element
Namespace (prefix)	nar
Name	rt
Title	Ruby Text
Definition	Ruby text.
User Note(s)	Also see **ruby** (page 240).
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)	

13.6.169 Same As

Table 181. Same As

(XML) Data Model	Element		
Namespace (prefix)	nar		
Name	sameAs {Relationship}		
Title	Same As		
Definition	An identifier of an equivalent concept.		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	**FlexPropType** (page 287)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **timeValidityAttributes** (page 307)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.170 SameAs for a Scheme

Table 182. Same As of a Scheme

(XML) Data Model	Element
Namespace (prefix)	nar
Name	sameAs {Scheme}
Title	Same As for a Scheme
Definition	A URI which identifies another scheme with concepts that use the same codes and are semantically equivalent to the concepts of this scheme.
User Note(s)	
Implementation Note(s)	
XML Schema Spec	At: Both CCL and PCL
Datatype	**IRIType** (page 291)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.171 Scheduled

Table 183. Scheduled

(XML) Data Model	Element
Namespace (prefix)	nar
Name	scheduled
Title	Scheduled
Definition	The intended time of delivery for the planned G2 item.
User Note(s)	MUST correspond to the itemClass property of the planned item.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	**ApproximateDateTimePropType** (page 267)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.172 Scheme Declaration

Table 184. 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 291); The URI which identifies the scheme.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
Child Element(s)	▪ **sameAs {Scheme}** (page 245) (0..unbounded)		
XML Schema Note(s)			
Example(s)			
Processing Model for the sameAs child	* The scheme identified by the @uri must not use a code which does not exist in all of the schemes identified by the sameAs elements. * The concept identified by a code in the scheme identified by the @uri must be semantically equivalent to the concept with the same code in all of the schemes identified by the sameAs elements.		

13.6.173 Sender

Table 185. 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)	If both are present the @literal and the property string value SHOULD be identical. If both are present but not identical @literal takes precedence		
XML Schema Spec	At: Both CCL and PCL		
Datatype	XML Schema string		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **qualifyAttributes** (page 309)	**Name**	**Datatype**
		qcode (0..1)	QCodeType
		literal (0..1)	XML Schema normalizedString
		type	QCodeType
		role	QCodeType
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.174 Signal

Table 186. 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 298)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ severity (0..1); **QCodeType** (page 297); Indicates how severe the impact from the signal is. The recommended vocabulary is the IPTC Severity NewsCodes http://cv.iptc.org/newscodes/severity/
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.175 Slugline

Table 187. 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: PCL		
Datatype	**IntlStringType** (page 290)		
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.		
	▪ role (0..1); **QCodeType** (page 297); The role this slugline plays in the scope of the full content.		
	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.176 Span

Table 188. 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 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
	▪ **editAttributes** (page 306)	**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)			

13.6.177 Start Date/Time

Table 189. 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 267)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.178 Subject

Table 190. 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 278)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **rankingAttributes** (page 307)	**Name**	**Datatype**
		rank (0..1)	XML Schema nonNegativeInteger
Child Element(s)			
XML Schema Note(s)			
Example(s)			

13.6.179 Time Delimiter

Table 191. Time Delimiter

(XML) Data Model	Element
Namespace (prefix)	nar
Name	timeDelim
Title	Time Delimiter
Definition	A delimiter for a piece of streaming media content expressed in various time formats..
User Note(s)	The time unit may take the following values, taken from an IPTC defined controlled vocabulary: - *timecode*: An SMPTE timecode containing a string encoded identification. Timestamp format: hh:mm:ss:ff (ff for frames). - *timeCodeDropFrame*: An SMPTE timecode containing a string encoded identification. Timestamp format: hh:mm:ss:ff (ff for frames). The drop frame flag should be set. - *editUnit*: The editUnit is the amount of time per video frame (1s / number of frames per second) or the amount of time per audio sample (1s / number of samples per second), for which the video frame rate or audio sample rate must be known. Timestamp format: long unsigned integer. - *normalPlayTime*: Indicates the position relative to the beginning of the presentation. Timestamp format: hh:mm:ss.mmm (mmm for milliseconds). See also: RFC 2326. - *seconds*: Time given in full seconds. Timestamp format: long unsigned integer. - *milliseconds*: Time given in full milliseconds. Timestamp format: long unsigned integer.
Implementation Note(s)	If a time unit IS NOT present, the value *editUnit* MUST be assumed. Any timestamps MUST be formatted appropriately for the time unit (as detailed above). All timestamps SHOULD be zero-padded from the left as applicable, e.g. a normalPlay-Time value starting at 12 seconds would be represented as '00:00:12.000'.
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	Mandatory IPTC NewsCodes: http://cv.iptc.org/newscodes/timeunit/
Attribute(s)	▪ start (1); XML Schema string; The timestamp corresponding to the start of the part.
	▪ end (1); XML Schema string; The timestamp corresponding to the end of the part.
	▪ timeunit (1); **QCodeType** (page 297); The unit used for the start and end time-stamps.
	▪ **editAttributes** (page 306)

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

Child Element(s)	

Table 191. Time Delimiter

XML Schema Note(s)	

Example(s)	`<!-- Example: Defining a 1 second part running from 2S to 3S, using 'seconds' -->`
	`<partMeta>`
	`<timeDelim start="2" end="3" timeunit="timeunit:seconds"/>`
	`</partMeta>`
	`<!-- Example: Defining a 1.5 second part running from 2.0S to 3.5S, using 'normalPlayTime' -->`
	`<partMeta>`
	`<timeDelim start="00:00:02.000" end="00:00:03.500" time-unit="timeunit:normalPlayTime"/>`
	`</partMeta>`

13.6.180 Timestamp

Table 192. Timestamp

(XML) Data Model	Element
Namespace (prefix)	nar
Name	timestamp
Title	Timestamp
Definition	A date plus a mandatory time with time zone 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 275)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); XML Schema string; A refinement of the semantics of the property. The string value may take a QCode. That the string should be interpreted as a QCode has to be defined outside of the G2 specification by the creator of the News Message.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.181 Transmission Identifier

Table 193. 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)	

13.6.182 Type of a Concept

Table 194. 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 298)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.183 Urgency

Table 195. 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 289)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 306)	**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)			

13.6.184 Usage Terms

Table 196. 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 301)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.6.185 Visual Region Delimiter

Table 197. 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.		
	▪ **editAttributes** (page 306)	**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)			

13.6.186 Web Address

Table 198. 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 291)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 295); A refinement of the semantics of the web address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.7 Element Group Definitions

13.7.1 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 199. Concept Definition Group Elements

Element Title	Element Name	Card	Described on Page
Concept Name	**name**	(0..unbounded)	91
Concept Definition	**definition**	(0..unbounded)	88
Note	**note**	(0..unbounded)	201
Facet	**facet**	(0..unbounded)	147
Remote Information	**remoteInfo**	(0..unbounded)	236
Hierarchy Info	**hierarchyInfo**	(0..unbounded)	

13.7.2 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 200. Concept Relationships Group Elements

Element Title	Element Name	Card	Described on Page
Same As	**sameAs {Relationship}**	(0..unbounded)	244
Broader	**broader**	(0..unbounded)	79
Narrower	**narrower**	(0..unbounded)	194
Related	**related**	(0..unbounded)	229

13.7.3 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 201. Entity Details Group Elements

Element Title	Element Name	Card	Described on Page
Person Details	**personDetails**	(1)	214
Organisation Details	**organisationDetails**	(1)	206
Geopolitical Area Details	**geoAreaDetails**	(1)	155
POI Details	**POIDetails**	(1)	218
Object Details	**objectDetails**	(1)	202

13.7.4 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 202. Administrative Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Urgency	**urgency**	(0..1)	259
Date Content Created	**contentCreated**	(0..1)	113
Date Content Modified	**contentModified**	(0..1)	114
Located	**located**	(0..unbounded)	175
Information Source	**infoSource**	(0..unbounded)	259
Creator	**creator**	(0..unbounded)	109
Contributor	**contributor**	(0..unbounded)	104
Audience	**audience**	(0..unbounded)	76
Excluded Audience	**exclAudience**	(0..unbounded)	144
Alternative Identifier	**altId**	(0..unbounded)	70

13.7.5 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 203. Knowledge Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Language	**language**	(0..unbounded)	185
Keyword	**keyword**	(0..unbounded)	183
Subject	**subject**	(0..unbounded)	253
Description	**description**	(0..unbounded)	130

13.7.6 Descriptive Metadata Core 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 204. Descriptive Metadata Core Group Elements

Element Title	Element Name	Card	Described on Page
Language	**language**	(0..unbounded)	185
Keyword	**keyword**	(0..unbounded)	183
Subject	**subject**	(0..unbounded)	253
Slugline	**slugline**	(0..unbounded)	250
Headline	**headline**	(0..unbounded)	160
Description	**description**	(0..unbounded)	130

13.7.7 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 205. Descriptive Metadata Group Elements

Element Title	Element Name	Card	Described on Page
Language	**language**	(0..unbounded)	185
Genre	**genre**	(0..unbounded)	153
Keyword	**keyword**	(0..unbounded)	183
Subject	**subject**	(0..unbounded)	253
Slugline	**slugline**	(0..unbounded)	250
Headline	**headline**	(0..unbounded)	160
Dateline	**dateline**	(0..unbounded)	127
By	**by**	(0..unbounded)	80
CreditLine	**creditline**	(0..unbounded)	110
Description	**description**	(0..unbounded)	130

13.7.8 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 206. Item Management Group Elements

Element Title	Element Name	Card	Described on Page
Item Class	**itemClass**	(1)	176
Content Provider	**provider**	(1)	102
Date Item Version Created	**versionCreated**	(1)	117
Date Item First Created	**firstCreated**	(0..1)	116
Date Item Embargo Ends	**embargoed**	(0..1)	115
Publish Status	**pubStatus**	(0..1)	225
Role in the Workflow	**role**	(0..1)	239
File Name	**filename**	(0..1)	149
Generator Tool	**generator**	(0..unbounded)	152
Profile	**profile**	(0..1)	223
Editorial Service	**service**	(0..unbounded)	137
Item Title	**title**	(0..unbounded)	182
Editorial Note	**edNote**	(0..unbounded)	136
Member Of	**memberOf**	(0..unbounded)	192
Instance Of	**instanceOf**	(0..unbounded)	173
Signal	**signal**	(0..unbounded)	249
Alternative Representation	**altRep**	(0..unbounded)	72
Deliverable Of	**deliverableOf**	(0..1)	131
Hash Value	**hash**	(0..unbounded)	131

13.8 Datatype Definitions

13.8.1 Any Item Type

Table 207. 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.
	▪ **i18nAttributes** (page 306) — **Name**: xml:lang (0..1), **Datatype**: XML Schema language; **Name**: dir (0..1), **Datatype**: XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ W3C XML Digital Signature (from namespace http://www.w3.org/2000/09/xmldsig#) (0..1)
	▪ **catalogRef** (page 232) (0..unbounded)
	▪ **catalog** (page 82) (0..unbounded)
	▪ **hopHistory** (page 163) (0..1)
	▪ **rightsInfo** (page 237) (0..unbounded)
	▪ **itemMeta** (page 178) (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)	

13.8.2 Approximate Date and Time Property Type

Table 208. 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 303); The date (and optionally time) at which the approximation range begins.
	▪ approxend (0..1); **TruncatedDateTimeType** (page 303); The date (and optionally time) at which the approximation range ends.
	▪ **editAttributes** (page 306) — **Name**: id (0..1), creator (0..1), modified (0..1); **Datatype**: XML Schema ID, QCodeType, DateOptTimeType
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.

13.8.3 Audience Type

Table 209. 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 280)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ significance (0..1); **Int1To9Type** (page 289). A qualifier which indicates the expected significance of the content for this specific audience.

▪ **quantifyAttributes** (page 309)	**Name**	**Datatype**
	confidence (0..1)	Int100Type
	relevance (0..1)	Int100Type
	why (0..1)	QCodeType

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

13.8.4 Block Type

Table 210. 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 295); 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.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **a** (page 73) (0..unbounded)		
	▪ **span** (page 251) (0..unbounded)		
	▪ **ruby** (page 240) (0..unbounded)		
	▪ **br** (page 186) (0..unbounded)		
	▪ **inline** (page 165) (0..unbounded)		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.8.5 Concept Identifier Type

Table 211. 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 297); A qualified code which identifies a concept.
	▪ created (0..1); **DateOptTimeType** (page 273); The date (and optionally the time) when the identifier was created.
	▪ retired (0..1); **DateOptTimeType** (page 273); 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)	

13.8.6 Content Metadata Type

Table 212. 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 306)	**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 264) (0..1)	**Element Name**	**Page**
		urgency (0..1)	259
		contentCreated (0..1)	113
		contentModified (0..1)	114
		located (0..unbounded)	189
		infoSource (0..unbounded)	175
		creator (0..unbounded)	109
		contributor (0..unbounded)	104
		audience (0..unbounded)	76
		exclAudience (0..unbounded)	144
		altId (0..unbounded)	70
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.8.7 Date and Optional Time Property Type

Table 213. 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 with 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 306)	**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)			

13.8.8 Date and Optional Time Type

Table 214. 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 with 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)	

13.8.9 Date and Time or Null Value Property Type

Table 215. Date and Time or Null Value Property Type

(XML) Data Model	Type		
Namespace (prefix)	nar		
Name	DateTimeOrNullPropType		
Title	Date and Time or Null Value Property Type		
Definition	The type of a property with date and time - or Nothing		
User Note(s)			
Implementation Note(s)			
XML Schema Spec	At: PCL		
Datatype	A union of the XML Schema dateTime type and an XML Schema string restricted to an empty value (UnionDateEmptyStringType).		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **editAttributes** (page 306)	**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)			

13.8.10 Date and Time Property Type

Table 216. 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 with 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 306)	**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)			

13.8.11 Electronic Address Type

Table 217. 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 295); A refinement of the semantics of the electronic address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.8.12 Electronic Address Tech Type

Table 218. 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 295); A refinement of the semantics of the electronic address.
	▪ tech (0..1); **QCodeType** (page 297); The technical variant of the electronic address.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.8.13 Flexible 1 Concept Property Type

Table 219. 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 280)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **quantifyAttributes** (page 309)	**Name**	**Datatype**
		confidence (0..1)	Int100Type
		relevance (0..1)	Int100Type
		why (0..1)	QCodeType
Child Element(s)	▪ **bag** (page 77) (0..1)		
XML Schema Note(s)			
Example(s)			

13.8.14 Flexible 1 Party Property Type

Table 220. 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 284)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ role (0..1); **QCodeListType** (page 295); A refinement of the semantics of the property.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.8.15 Flexible 1 Property Type

Table 221. 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 297); A qualified code assigned as a property value. (For an inlineRef element this is the identifier of the described concept.) Or ▪ literal (0..1); XML Schema normalizedString; A free-text value assigned as a property value. (For an inlineRef element this is the identifier of the described concept.)		
	▪ type (0..1); **QCodeType** (page 297); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	**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 263) (0..1)	**Element Name**	**Page**
		name (0..unbounded)	91
		definition (0..unbounded)	88
		facet (0..unbounded)	147
		remoteInfo (0..unbounded)	236
		note (0..unbounded)	201
	▪ **Concept Relationships Group** (page 263) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	79
		narrower (0..unbounded)	194
		related (0..unbounded)	229
		sameAs {Relationship} (0..unbounded)	244
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.8.16 Flexible Location Property Type

Table 222. 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 297); 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 297); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	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 263) (0..1)	**Element Name**	**Page**
		name (0..unbounded)	91
		definition (0..unbounded)	88
		facet (0..unbounded)	147
		remoteInfo (0..unbounded)	236
		note (0..unbounded)	201
	▪ **Concept Relationships Group** (page 263) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	79
		narrower (0..unbounded)	194
		related (0..unbounded)	229
		sameAs {Relationship} (0..unbounded)	244
	▪ **geoAreaDetails** (page 155) (0..1) Or ▪ **POIDetails** (page 218) (0..1)		
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		

Table 222. Flexible Location Property Type (Continued)

XML Schema Note(s)	
Example(s)	

13.8.17 Flexible Organisation Property Type

Table 223. 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 297); 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 297); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **organisationDetails** (page 206) (0..1)		
	▪ **Concept Definition Group** (page 263) (0..1)	**Element Name**	**Page**
		name (0..unbounded)	91
		definition (0..unbounded)	88
		facet (0..unbounded)	147
		remoteInfo (0..unbounded)	236
		note (0..unbounded)	201
	▪ **Concept Relationships Group** (page 263) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	79
		narrower (0..unbounded)	194
		related (0..unbounded)	229
		sameAs {Relationship} (0..unbounded)	244
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.8.18 Flexible Party Property Type

Table 224. 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 297); 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 297); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	Name	Datatype
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.

Table 224. Flexible Party Property Type (Continued)

Child Element(s)	▪ **personDetails** (page 214) (0..1) or ▪ **organisationDetails** (page 206) (0..1)		
	▪ **Concept Definition Group** (page 263) (0..1)	**Element Name**	**Page**
		name (0..unbounded)	91
		definition (0..unbounded)	88
		facet (0..unbounded)	147
		remoteInfo (0..unbounded)	236
		note (0..unbounded)	201
	▪ **Concept Relationships Group** (page 263) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	79
		narrower (0..unbounded)	194
		related (0..unbounded)	229
		sameAs {Relationship} (0..unbounded)	244
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.8.19 Flex Person Property Type

Table 225. 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 297); 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 297); The type of the concept assigned as controlled or uncontrolled property value.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **personDetails** (page 214) (0..1)		
	▪ **Concept Definition Group** (page 263) (0..1)	**Element Name**	**Page**
		name (0..unbounded)	91
		definition (0..unbounded)	88
		facet (0..unbounded)	147
		remoteInfo (0..unbounded)	236
		note (0..unbounded)	201
	▪ **Concept Relationships Group** (page 263) (0..1)	**Element Name**	**Page**
		broader (0..unbounded)	79
		narrower (0..unbounded)	194
		related (0..unbounded)	229
		sameAs {Relationship} (0..unbounded)	244
	▪ Extension Point (0..unbounded). Any set of provider-defined properties.		
XML Schema Note(s)			
Example(s)			

13.8.20 Flexible Property Type

Table 226. Flexible Property Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	FlexPropType
Title	Flexible 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 297); 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 297); The type of the concept assigned as a controlled or uncontrolled property value.
	▪ **editAttributes** (page 306)

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

▪ **i18nAttributes** (page 306)

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

Child Element(s)	▪ **name** (page 224) (0..unbounded) ▪ **hierarchyInfo** (page 161) (0..unbounded)
XML Schema Note(s)	
Example(s)	

13.8.21 Integer 0 to 100 Type

Table 227. 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)	

13.8.22 Integer 1 to 9 Type

Table 228. 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)	

13.8.23 International String Type

Table 229. 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 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
	▪ **editAttributes** (page 306)	**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)			

13.8.24 IRI Type

Table 230. 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)	

13.8.25 Label 1 Type

Table 231. 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 295); A refinement of the semantics of the label.		
	▪ media (0..1); XML Schema NMTOKENS; An indication of the target media type(s), values as defined by the Cascading Style Sheets (CSS) specification.		
	▪ **editAttributes** (page 306)	**Name**	**Datatype**
		id (0..1)	XML Schema ID
		creator (0..1)	QCodeType
		modified (0..1)	DateOptTimeType
	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **a** (page 73) (0..unbounded)		
	▪ **span** (page 251) (0..unbounded)		
	▪ **ruby** (page 240) (0..unbounded)		
	▪ **inline** (page 165) (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)			

13.8.26 Link 1 Type

Table 232. Link 1 Type

(XML) Data Model	Type
Namespace (prefix)	nar
Name	Link1Type
Title	Link 1 Type
Definition	A datatype for linking this item to other items or resources.
User Note(s)	To identify the target resource either the residref attribute or the href attribute MUST be set, optionally both MAY be used in parallel. The residref attribute identifies the target resource by its globally unique identifier (if the resource has such an identifier), while the href attribute identifies the location of the target resource in e.g. a (remote) file system. If the target resource is an Item and the residref attribute is used, a version attribute MAY indicate the target Item version; in the absence of version information, the target resource is the latest version available.
Implementation Note(s)	
XML Schema Spec	At: PCL
Datatype	
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ rank (0..1); XML Schema nonNegativeInteger; The rank of the link among other links.
	▪ rel (0..1); **QCodeType** (page 297); The identifier of the relationship between the current Item and the target resource.

▪ **editAttributes** (page 306)

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

▪ **i18nAttributes** (page 306)

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

▪ **timeValidityAttributes** (page 307)

Name	DataType
validfrom (0..1)	DateOptTimeType
validto (0..1)	DateOptTimeType

▪ **targetResourceAttributes** (page 311)

Name	Datatype
href (0..1)	IRIType
residref (0..1)	XML Schema string
version (0..1)	XML Schema positiveInteger
contenttype (0..1)	XML Schema string
format (0..1)	QCodeType
size (0..1)	XML Schema non NegativeInteger

▪ guidref (0..1); XML Schema string; The use of this attribute is DEPRECATED, use *residref* instead.

▪ hreftype (0..1); XML Schema normalizedString; The use of this attribute is DEPRECATED, use *contenttype* instead.

Table 232. Link 1 Type (Continued)

Child Element(s)	▪ Hint and Extension Point (0..unbounded). Properties from the NAR namespace or from another XML namespace may be added. The purpose of properties from the NAR namespace is to add a set of hints, i.e. properties which have to comply with the structure of the G2 item target resource but do not have to be extracted from it. These properties must be added this way: - Immediate child properties of <itemMeta>, <contentMeta>, or <concept> - optionally with their descendants - may be used directly under the extension point - All other properties require the full path excluding only the item's root element.
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)	

13.8.27 QCode List Type

Table 233. 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)	

13.8.28 QCode Property Type

Table 234. 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 297); A qualified code assigned as a property value.		
	▪ **editAttributes** (page 306)	**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)			

13.8.29 QCode Type

Table 235. 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)	

13.8.30 Qualified Property Type

Table 236. 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 296)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ **i18nAttributes** (page 306)	**Name**	**Datatype**
		xml:lang (0..1)	XML Schema language
		dir (0..1)	XML Schema string: enumeration *ltr*, *rtl*.
Child Element(s)	▪ **name** (page 91) (0..unbounded) ▪ **hierarchyInfo** (page 161) (0..unbounded)		
XML Schema Note(s)			
Example(s)			

13.8.31 Recurrence Rule Type

Table 237. 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 273); 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.
	▪ bysecond (0..1); tokens of XML Schema nonNegativeInteger 0..59 ; The BYSECOND rule part specifies a space separated list of seconds within a minute.
	▪ byminute (0..1); tokens of XML Schema nonNegativeInteger 0..59; The BYMINUTE rule part specifies a space separated list of minutes within an hour.
	▪ byhour (0..1); tokens of XML Schema nonNegativeInteger 0..23; The BYHOUR rule part specifies space separated list of hours of the day.
	▪ byday (0..1); tokens of XML Schema string, regEx pattern = "(\-|\+)?([0-9]){0,2}(MO|TU|WE|TH|FR|SA|SU)": ; The BYDAY rule part specifies a space separated list of days of the week.
	▪ bymonthday (0..1); tokens of XML Schema Integer -31..-1 and 1..31; The BYMONTHDAY rule part specifies a space separated list of days of the month.
	▪ bymonth (0..1); tokens of XML Schema nonNegativeInteger 1..12; The BYMONTH rule part specifies a space separated list of months of the year.
	▪ byyearday (0..1); tokens of XML Schema Integer -366..-1 and 1..366; The BYYEARDAY rule part specifies a space separated list of days of the year.
	▪ byweekno (0..1); tokens of XML Schema Integer -53..-1 and 1..53; The BYWEEKNO rule part specifies a space separated list of ordinals specifying weeks of the year.
	▪ bysetpos (0..1); tokens of XML Schema Integer -366..-1 and 1..366; The BYSETPOS rule part specifies a space separated list of values which corresponds to the nth occurrence within the set of events specified by the rule.
	▪ wkst (0..1); XML schema string, enumeration; The day on which the workweek starts.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.8.32 Related Concept Type

Table 238. 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 **FlexPropType** (page 287)		
Internally Ctrl Values			
Externally Ctrl Values			
Attribute(s)	▪ rel (0..1); **QCodeType** (page 297); The identifier of the relationship between the current concept and the target concept.		
	▪ rank (0..1); XML Schema positiveInteger; The rank of the current concept among concepts having a relationship to the target concept.		
	▪ **timeValidityAttributes** (page 307)	**Name**	**DataType**
		validfrom (0..1)	DateOptTimeType
		validto (0..1)	DateOptTimeType
Child Element(s)	▪ **facet** (page 147) (0..unbounded)		
XML Schema Note(s)			
Example(s)			

13.8.33 Rights Label Type

Table 239. 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	**BlockType** (page 269)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ href (0..1); **IRIType** (page 291); The locator of a remote expression of rights.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.8.34 Truncated Date and Time Property Type

Table 240. 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. If the time part is present the time zone SHOULD NOT be omitted.		
User Note(s)			
Implementation Note(s)	TruncatedDateTimePropType is used as a property datatype. Values my look like that: YYYY-MM-DD"T"hh:mm:ss.sssTZ YYYY-MM-DD"T"hh:mm:ssTZ YYYY-MM-DD YYYY-MM YYYY		
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 306)	**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)			

13.8.35 Truncated Date and Time Type

Table 241. 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. If the time part is present the time zone SHOULD NOT be omitted.
User Note(s)	
Implementation Note(s)	TruncatedDateTimeType is used as a qualifier datatype. Values my look like that: YYYY-MM-DD"T"hh:mm:ss.sssTZ YYYY-MM-DD"T"hh:mm:ssTZ YYYY-MM-DD YYYY-MM YYYY
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)	

13.8.36 Typed Qualified Property Type

Table 242. 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 298)
Internally Ctrl Values	
Externally Ctrl Values	
Attribute(s)	▪ type (0..1); **QCodeType** (page 297); The type of the concept assigned as property value.
Child Element(s)	
XML Schema Note(s)	
Example(s)	

13.8.37 Versioned String Type

Table 243. 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 290)
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)	

13.9 Attribute Group Definitions

13.9.1 Internationalization Attributes Group

Table 244. 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.

13.9.2 Editing Attributes Group

These attributes are used only at PCL.

Table 245. 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.

13.9.3 Persistent Editing Attributes

Table 246. persistentEditAttributes

Title	Name	Card	Datatype	Definition
Local Identifier	id	0..1	XML Schema ID	The local identifier of the element which MUST be persistent for all versions of the item, i.e. for its entire lifecycle.
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.

13.9.4 Time Validity Attributes Group

These attributes are used only at PCL.

Table 247. 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.

13.9.5 Ranking Attributes Group

These attributes are used only at PCL.

Table 248. rankingAttributes

Title	Name	Card	Datatype	Definition
Rank	rank	0..1	XML Schema nonNegativeInteger	Indicates the relative importance of properties in a list. It applies only to set of properties with the same name. See note on rules for ranking below..

Note: Currently there is only one attribute in this group by design; this group may be extended in the future.

Processing rules for the *rank* attribute:

Properties with a lower value of the rank attribute have a higher importance than properties with a higher value of this attribute. All properties with the same value of the rank attribute have the same importance. All properties without a rank attribute have the same importance, which is lower than the importance of all properties with this attribute.

If relative importance is being used to determine display order, then:

- Properties with a lower value of the rank attribute should be displayed before properties with a higher value of this attribute.

- Properties with the same value of the rank attribute should be ordered within this rank alphabetically by their names if these are available. If some or all of the names are available in multiple languages, the order of the properties will depend on the language chosen.

- All properties without a rank attribute should be displayed after all properties with this attribute.

Examples (using rank with the language property):

```
<!-- Rank as: all equal (implicit) -->
<language tag="en"/>
<language tag="fr"/>
<language tag="es"/>
<language tag="de"/>
<!-- Rank as: en, then any others -->
<language tag="en" rank="1"/>
<language tag="fr"/>
<language tag="es"/>
<language tag="de"/>
<!-- Rank as: en, then fr, then es, then de -->
<language tag="en" rank="1"/>
<language tag="fr" rank="2"/>
<language tag="es" rank="3"/>
<language tag="de" rank="4"/>
<!-- Rank as: en, then fr, then any others -->
<language tag="en" rank="1"/>
<language tag="fr" rank="2"/>
<language tag="es"/>
<language tag="de"/>
<!-- Rank as: en and fr, then any others -->
<language tag="en" rank="1"/>
<language tag="fr" rank="1"/>
<language tag="es"/>
<language tag="de"/>
```

13.9.6 Qualify Attributes Group

Table 249. qualifyAttributes

Title	Name	Card	Datatype	Definition
QCode	qcode	0..1	**QCodeType** (page 297)	A qualified code assigned as property value.
Literal	literal	0..1	XML Schema normalizedString	A free-text value assigned as property value.
Type	type	0..1	**QCodeType** (page 297)	The type of the concept assigned as a controlled or an uncontrolled property value.
Role	role	0..1	**QCodeType** (page 297)	A refinement of the semantics of the property.

13.9.7 Quantify Attributes Group

These attributes are used only at PCL.

Table 250. 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 obtained 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).

13.9.8 News Content Attributes

Table 251. newsContentAttributes

Title	Name	Card	Datatype	Definition
Local Identifier	id	0..1	XML Schema ID	The local identifier of the element which MUST be persistent for all versions of the item, i.e. for its entire lifecycle.
Rendition	rendition	0..1	QCodeType	The specific rendition of content this component represents. A specific value for rendition MUST NOT be used more than once for elements under contentSet of a NewsItem.
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.
- Note that *contentype* and *itemClass* in **Item Metadata** (page 178) 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 (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”.

13.9.9 Target Resource Attributes Group

Table 252. targetResourceAttributes

Title	Name	Card	DataType	Definition
Hyperlink	href	0..1	IRIType	The locator of the target resource
Resource Identifier Reference	residref	0..1	XML Schema string	The provider's identifier of the target resource
Item Version	version	0..1	XML Schema positiveInteger	The version of the target resource
Persistent Id Reference	persistidref	0..1	XML Schema string	Points to an element inside the target resource which must be identified by an ID attribute having a value which is persistent for all versions of the target resource, i.e. for its entire lifecycle.
Content Type	contenttype	0..1	XML Schema string	The IANA (Internet Assigned Numbers Authority) MIME type of the target resource
Format	format	0..1	QCodeType	A refinement of a generic content type (i.e. IANA MIME type)
Size	size	0..1	XML Schema non NegativeInteger	The size in bytes of the target resource

13.9.10 News Content Characteristics

To be implemented as an attribute group.

Table 253. 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 Image Width for visual content.
Image Width Unit	widthunit	0..1	QCodeType	If present it defines the Width Unit for the Image Width.
Image Height	height	0..1	XML Schema nonNegativeInteger	The Image Height for visual content.
Image Height Unit	heightunit	0..1	QCodeType	If present it defines the Height Unit for the Image Height.
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 255**.
Image Colour Space	colourspace	0..1	QCodeType	The colour space of an image. Applies to image content.

Table 253. newsContentCharacteristics (Continued)

Title	Name	Card	Datatype	Definition
Colour Indicator	colourindicator	0..1	QCodeType	Indicates whether the still or moving image is coloured or black and white. The recommended vocabulary is the IPTC Colour Indicator NewsCodes http://cv.iptc.org/newscodes/colourindicator/
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 time units defined by durationUnit. The default time unit is seconds. Applies to audio-visual content.
Unit of Duration	durationunit	0..1	QCodeType	If present it defines the time unit for the duration attribute. Only codes for integer value time units of the recommended CV (available at http://cv.iptc.org/newscodes/timeunit/) must be applied.
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 bps. Indicates the average variable bit rate if audiovbr is set to true.
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 bit rate in bps. Indicates the average variable bit rate if videovbr is set to true.
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.
Video Frame Rate	videoframerate	0..1	XML Schema decimal	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.

Table 253. newsContentCharacteristics (Continued)

Title	Name	Card	Datatype	Definition
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.
Video Scaling	videoscaling	0..1	QCodeType	Indicates how the original content was scaled to this format. The recommended vocabulary is the IPTC Video Scaling NewsCodes http://cv.iptc.org/news-codes/videoscaling/
Video Defintion	videodefinition	0..1	QCodeType	Indicates which video definition is applied to this rendition of video content but it does not imply any particular technical characteristics of the video.The recommended vocabulary is the IPTC Video Definition NewsCodes http://cv.iptc.org/newscodes/videodefinition/

Table 254 defines the default units to be used for the Image Width and/or Image Height, if the corresponding Image Width Unit and/or Image Height Unit are not specified.

Table 254. Default Image Height/Width Unit Values

Content Type	Height Unit (default)	Width Unit (default)
Picture	Pixels	Pixels
Graphic: Still / Animated	Points	Points
Video (Analog)	Lines	Pixels
Video (Digital)	Pixels	Pixels

Table 255 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 255. 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.	
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.	

14 Glossary

Table 256. 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”.
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.

Table 256. Glossary (Continued)

Term	Definition
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 256. 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 {Relationship}** (page 244) 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**.

15 References

Table 257. References

Subject	Description
IPTC Documents	
NML-BR	IPTC NewsML 2 Business Requirements: http://www.iptc.org/std/NewsML/2.0/specification/NewsML_2.0-spec-BusinessRequirements_1.pdf
EventsML-G2	Specifications for EventsML-G2: http://www.iptc.org/std/NewsML-G2/2.9/specification/
NewsML-G2	Specifications for NewsML-G2: http://www.iptc.org/std/NewsML-G2/2.9/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 XSD	W3C XML Schema 1.0 specifications at: 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://www.rfc-editor.org/rfc/bcp/bcp47.txt
iCalendar	iCalendar as specified by RFC 2445: http://www.ietf.org/rfc/rfc2445.txt

16 Index

16.1 Elements

16.2 Datatypes

Contact Information

Contact the IPTC by:

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

Web:

www.iptc.org

Twitter: @IPTC

Email:

office@iptc.org